

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. ІГОРЯ СІКОРСЬКОГО»

ІНФОРМАТИКА

КОНСПЕКТ ЛЕКЦІЙ

для студентів денної форми навчання спеціальності 131 «Прикладна механіка» спеціалізації «Машини і технології пакування» і спеціальності 133 «Галузеве машинобудування» спеціалізації «Інжиніринг, обладнання та технології виробництв полімерних та будівельних матеріалів і виробів»

Рекомендовано Вченою радою інженерно-хімічного факультету

Київ
НТУУ «КПІ»
2017

Конспект лекцій з дисципліни «Інформатика» для студ. денної форми навчання спеціальності 131 «Прикладна механіка» спеціалізації «Машини і технології пакування» і спеціальності 133 «Галузеве машинобудування» спеціалізації «Інжиніринг, обладнання та технології виробництв полімерних та будівельних матеріалів і виробів» / Уклад.: І.О. Казак. – К.: НТУУ «КПІ ім. Ігоря Сікорського», 2017. – 106 с.

*Гриф надано Вченою радою ІХФ
(Протокол № 2 від 23.03. 2017 р.)*

Навчальне видання

ІНФОРМАТИКА

Конспект лекцій для студентів денної форми навчання спеціальності 131 «Прикладна механіка» спеціалізації «Машини і технології пакування» і спеціальності 133 «Галузеве машинобудування» спеціалізації «Інжиніринг, обладнання та технології виробництв полімерних та будівельних матеріалів і виробів»

Авторська редакція

Укладач:

І.О. Казак, к.п.н.

Відповідальний редактор

Д.Е. Сідоров, к.т.н., доц.

Рецензент:

А.Р. Степанюк, к.т.н., доц.

ЗМІСТ

Вступ.....	4
Лекція 1. Структурування даних: складання лінійних структур, таблиць, матриць. Ієрархічні структури даних. Файли та файлова структура даних: одиниці вимірювання даних та одиниці зберігання даних.....	5
Лекція 2. Система збереження інформації в комп'ютері. Типи і види файлів. Робота з файлами. Макетування документів в Word.....	9
Лекція 3. Кодування чисел двійковим кодом. Властивості двійкової системи кодування даних Різноманітність кодування текстових даних, графічних даних, звукових даних. Кількість інформації.....	13
Лекція 4. Особливості автоматичних розрахунків на ПЕОМ.....	21
Лекція 5. Поняття про операційну систему ПЕОМ. Програмні продукти ПЕОМ. Система Windows.....	27
Лекція 6. Побудова графіків та діаграм редактором Excel. Розв'язання рівнянь чисельними методами; метод підбору за допомогою операції відділення кореню.....	31
Лекція 7. Розв'язання математичних задач на базі лінійних та нелінійних рівнянь за допомогою Excel.....	34
Лекція 8. Принципи математичних обчислень у редакторі MathCad.....	42
Лекція 9. Розв'язання лінійних та нелінійних рівнянь у MathCad наближеними методами (метод хорд і метод дотичних).....	45
Лекція 10. Розв'язання рівнянь у MathCad з застосуванням вбудованих функцій.....	51
Лекція 11. Операції з матрицями та векторами. Ранжовані змінні.....	56
Лекція 12. Побудова графіків; графічне зображення у MathCad матриць; тривимірні графіки.....	58
Лекція 13. Символьні обчислення MathCad. Символьне розв'язання лінійних та нелінійних рівнянь.....	62
Лекція 14. Елементи автоматичних обчислень у MathCad. Програмування математичних обчислень на вбудованій мові MathCad	66
Лекція 15. Основні принципи програмування в середовищі VBA. Створення макросів. Типи даних.....	73
Лекція 16. Математичні вирази. Операції вводу-виводу в середовищі VBA.....	75
Лекція 17. Організація умовних операторів і циклів в середовищі VBA	77
Лекція 18. Робота з масивами в середовищі VBA	79
Лекція 19. Алгоритми. Види алгоритмів. Написання і графічне зображення алгоритмів.....	80
Лекція 20. Робота в електронному Кампусі НТУУ «КПІ ім. І.Сікорського»	83
Лекція 21. Створення презентацій в MS PowerPoint	96
Список рекомендованої літератури.....	106

Вступ

Дисципліна «Інформатика» вивчається студентами денної форми навчання на першому році навчання за освітньо-кваліфікаційним рівнем бакалавра зі спеціальності 131 «Прикладна механіка» спеціалізації «Машини і технології пакування» і зі спеціальності 133 «Галузеве машинобудування» спеціалізації «Інжиніринг, обладнання та технології виробництв полімерних та будівельних матеріалів і виробів» за рахунок аудиторних годин та часу, який відведено робочим навчальним планом з «Інформатики» на самостійну роботу. Зміст конспекту лекцій відповідає робочій програмі з кредитного модуля дисципліни «Інформатика».

Тому основна мета конспекту лекцій з «Інформатика» – це забезпечення теоретичними основами, а також методична підтримка навчальної і самостійної роботи студентів з курсу «Інформатика» зі спеціальності 131 «Прикладна механіка» спеціалізації «Машини і технології пакування» і зі спеціальності 133 «Галузеве машинобудування» спеціалізації «Інжиніринг, обладнання та технології виробництв полімерних та будівельних матеріалів і виробів», що полегшить опанування студентами даної дисципліни.

Лекція 1

Тема: Структурування даних: складання лінійних структур, таблиць, матриць. Ієрархічні структури даних. Файли та файлова структура даних: одиниці вимірювання даних та одиниці зберігання даних

«Інформатика» походить від французьких слів *information* (інформація) і *automatique* (автоматика) і дослівно означає «інформаційна автоматика». Англomовний варіант цього терміну – «*Computer science*», що означає «комп'ютерна наука».

Інформатика – це дисципліна, яка використовує комп'ютерну техніку для вивчення структури і загальні властивості інформації, а також закономірності та методи її створення, зберігання, пошуку, перетворення, передачі і застосування в різних сферах людської діяльності.

Інформатика охоплює такі галузі наукової діяльності:

- розробка, створення, використання та обслуговування систем обробки інформації, (включаючи комп'ютери та їх програмне забезпечення);
- організаційні, комерційні, адміністративні та соціально-політичні проблеми пов'язані з комп'ютеризацією; тобто як масове впровадження комп'ютерної техніки впливає на всі сфери життя людей.

Таким чином, інформатика базується на комп'ютерній техніці і без неї не має сенсу.

Інформатика – наукова дисципліна. *Основні напрямки розвитку інформатики:*

1. розробка обчислювальних систем і програмного забезпечення;
2. теорія інформації, що вивчає процеси, пов'язані з передачею, прийомом, перетворенням і зберіганням інформації;
3. методи штучного інтелекту, що дозволяють створювати програми для вирішення завдань, що вимагають певних інтелектуальних зусиль при виконанні їх людиною (логічний висновок, навчання, розуміння мови, візуальне сприйняття, ігри та ін);
4. системний аналіз, що полягає в аналізі призначення проектованої системи і у встановленні вимог, яким вона повинна відповідати;
5. методи машинної графіки, анімації, засоби мультимедіа;
6. засоби телекомунікації, в тому числі, глобальні комп'ютерні мережі, об'єднуючі все людство в єдине інформаційне співтовариство;
7. різноманітні прикладання, що охоплюють виробництво, науку, освіту, медицину, торгівлю, сільське господарство і всі інші види господарської та громадської діяльності.

Інформатику зазвичай представляють з двох частин:

1. технічні засоби;
 2. програмні засоби.
1. *Технічні засоби*, тобто апаратура комп'ютерів, в англійській мові позначаються словом *Hardware*, яке буквально перекладається як «тверді вироби», або в побуті – «залізо».
 2. А для *програмних засобів* вибрано дуже вдале слово *Software* (в перекладі з англійської мови буквально – «м'які вироби»), яке підкреслює рівнозначність програмного забезпечення та самої машини, і разом з тим підкреслює здатність програмного забезпечення модифікуватися, пристосовуватися, розвиватися.

Крім цих двох загальноприйнятих частин інформатики виділяють ще одну – *алгоритмічні засоби*.

Для неї використовується назва «*Brainware*» (від англ. «*Brain*» – інтелект). Ця гілка пов'язана з розробкою алгоритмів і вивченням методів та прийомів їхньої побудови.

Алгоритми – це правила, що пропонують виконання послідовностей дій, що призводять до вирішення завдання.

Не обов'язково алгоритм – це математика. Це перелік команд, за якими потрібно виконати деяку послідовність дій.

Приклад:

Алгоритм переходу вулиці: 1) «подивися наліво, переконайся, що ліворуч немає транспорту»; 2) «переходи вулицю»; 3) «дійшовши до середини, подивися направо, переконайся, що праворуч немає транспорту»;

4) «продовжуй переходити вулицю».

В термін інформація вкладають різний зміст у техніці, науці і в життєвих ситуаціях. Його не можна пояснити однією фразою.

У побуті ми називаємо інформацією будь-які дані або відомості, які когось цікавлять.

Наприклад, повідомлення про які-небудь події, про чийсь діяльність і т.п. *«Інформувати»* в цьому сенсі означає *«повідомити щось, невідоме раніше»*.

Інформація – це набір різних даних у вигляді малюнків, схем, словесного опису і т.д. про об'єкти і явища, їх параметрах, властивості і стан, Важливо, щоб ці дані могли б бути сприйняті суб'єктами життєдіяльності (живими організмами, управляючими машинами та ін.).

Одне і те ж інформаційне повідомлення (стаття в газеті, оголошення, лист, телеграма, довідка, розповідь, креслення, радіопередача і т.п.) може містити різну кількість інформації для різних людей – в залежності від їх попередніх знань, від рівня розуміння цього повідомлення і інтересу до нього. В принципі, деякий набір даних взагалі може не являти собою ніякої інформації. Так, повідомлення, складене іноземною мовою, не несе ніякої інформації людині, яка не знає цієї мови. Ніякої інформації не містить і повідомлення на знайомою мовою, якщо його зміст незрозумілий або вже відомо.

Тобто можна сказати, що інформація може бути, або може бути відсутньою або наявною (відповідно крайні випадки – «0» або «1»).

Тобто для того, щоб дані стали інформацією, суб'єкт інформаційного обміну повинен сприймати адекватно отримані відомості. В іншому випадку можна говорити про те, що інформація відсутня. Суб'єктивність і об'єктивність інформації.

Звідси випливає важливий висновок про те, що інформація не є характеристикою даних або повідомлення, а являє собою продукт взаємодії повідомленням і його споживачем. Без наявності споживача, говорити про інформацію безглуздо.

У випадках, коли говорять про автоматизовану роботу з інформацією за допомогою будь-яких технічних пристроїв, зазвичай в першу чергу цікавляться не змістом повідомлення, а тим, скільки символів це повідомлення містить.

Якщо мова йде про комп'ютерну техніку, то під інформацією розуміють деяку послідовність символічних позначень (букв, цифр, закодованих графічних образів і звуків і т.п.), в зрозумілому для комп'ютера вигляді. Якщо при цьому набір зазначених символів є випадковим і не має смислового навантаження, то він не є інформацією для комп'ютерних засобів.

Інформація в техніці може існувати в найрізноманітніших формах:

- у вигляді текстів, малюнків, креслень, фотографій;
- у вигляді світлових або звукових сигналів;
- у вигляді радіохвиль;
- у вигляді електричних та нервових імпульсів;
- у вигляді магнітних записів.

Необхідно відзначити, що властивостями інформації володіють як матеріальні предмети, процеси і явища, так і нематеріальні (поняття, характеристики, слова і т.п.) Всі вказані параметри називаються *інформаційними об'єктами*.

Інформація являє собою деяке повідомлення (набір даних). Це повідомлення передається у вигляді деяких сигналів (звукових, електричних, світлових) від деякого джерела інформації до її приймача по деякому каналу зв'язку між ними. Як правило, повідомлення кодується джерелом в сигнал, що передається. Цей сигнал посиляється по каналу зв'язку. Приймач приймає сигнал, який декодується і стає прийнятим повідомленням.

Приклади:

1. Повідомлення, що містить інформацію про погоду, передається приймачу (телеглядачеві) від джерела – фахівця-метеоролога допомогою каналу зв'язку - телевізійної передавальної апаратури і телевізора;

2. Приймач інформації (телеглядач) своїми органами почуттів (око, вухо) сприймає інформацію із зовнішнього світу, переробляє її в певну послідовність нервових імпульсів, передає імпульси по нервових волокнах, зберігає в пам'яті у вигляді стану нейронних структур мозку, відтворює у вигляді звукових сигналів, рухів і т.п., використовує в процесі своєї життєдіяльності.

3. Звичайний розмова між співрозмовниками (інформаційний обмін) відбувається за тими ж принципами. Якщо один зі співрозмовників (джерело інформації) хоче розповісти про щось слухачеві (приймач інформації), він кодує це поняття певним поєднанням звукових сигналів (словом).

Дії, які можна виконувати з інформацією:

- | | |
|----------------------|--------------------|
| • поширювати; | • передавати; |
| • сприймати; | • перетворювати; |
| • комбінувати; | • використовувати; |
| • запам'ятовувати; | • обробляти; |
| • ділити на частини; | • приймати; |
| • копіювати; | • спрощувати; |
| • збирати; | • зберігати; |
| • шукати; | • вимірювати; |
| • руйнувати; | • та ін. |

Властивості інформації:

- достовірність;
- повнота;
- цінність;
- своєчасність;
- зрозумілість;
- доступність;
- стислість;
- та ін.

Так як ми говоримо про науку «Інформатика», то необхідно ввести деякі кількісні характеристики поняття «інформація». Отже, питання: чи можна кількісно виміряти інформацію? Для чого це потрібно? Справа в тому, що в багатьох практичних випадках якісними особливостями інформації можна знехтувати.

Приклад:

Необхідно передати дані про студентів першого курсу, присутніх на лекції з інформатики в деканат. Як це зробити?

1 *спосіб*. Викладач може сфотографувати аудиторію і передати це зображення по комп'ютерній мережі на сервер деканату;

2 *спосіб*. Викладач може написати список студентів, може зазначити тільки присутніх або тільки відсутніх і т.п.

Результат може бути один: деканат зробить висновок з отриманої інформації і, наприклад, зніме стипендію відсутнім на лекції.


Тобто зі сказаного випливає, що найважливішим в інформаційному процесі є отримання **ВИСНОВКУ**: в якості *одиниці інформації* умовилися прийняти *один біт* (з англ. *bit* – «*binary, digit*» – двійкова цифра).

Один біт виражає два поняття: 0 або 1 («ні» або «так»):


1-ше поняття: «ні» або 0
2 поняття: «так» або 1 = 1 біт

Необхідно зазначити, що 1 біт = одна цифра (0 або 1), тобто можна вважати, що 1 біт – це «1» і 1 біт – це «0». У той же час необхідно розуміти, що 1 біт – це 2 поняття.

Два біта, тобто дві цифри в різних поєднаннях, відповідають 4 поняттям:

2 біта

00 01 10 11
Два розряди

Три біта, тобто три цифри в різних поєднаннях, відповідають 8 поняттям:

3 біта

000 001 010 100 101 110 011 111
Три розряди

Якщо присвоїти кожному з поєднань трьох цифр яке-небудь поняття, то можна закодувати певну інформацію. Зазвичай при вживанні двійкового кодування використовується поняття розрядності кодування: використання двох цифр відповідає двохраньковому кодуванню, використання трьох цифр - трьоххраньковому. Відзначимо особливість впливу розрядності кодування на кількість понять, які можна закодувати: збільшення на одиницю розрядності (або що те ж саме – кількості бітів) подвоює кількість кодованих понять.

Загальна формула, яка виражає співвідношення між кодованими поняттями і розрядністю кодування (тобто кількістю бітів, які використовуються для кодування), має вигляд:

$$N = 2^m,$$

де N - кількість незалежних кодованих значень (стільки понять можна закодувати різними бітами);

m - розрядність двійкового кодування (кількість бітів).

Біт – занадто дрібна одиниця виміру. На практиці частіше застосовується більш велика одиниця - *байт*, дорівнює восьми бітам. Саме вісім бітів потрібно для того, щоб закодувати будь-який з 256 символів алфавіту клавіатури комп'ютера ($256=2^8$).

Широко також використовуються ще більші похідні одиниці інформації:

- 1 Кілобайт (кбайт) = 1024 байт = 2^{10} байт,
- 1 Мегабайт (Мбайт) = 1024 Кбайт = 2^{20} байт,
- 1 Гігабайт (ГБ) = 1024 Мбайт = 2^{30} байт.

Останнім часом у зв'язку зі збільшенням обсягів оброблюваної інформації входять у вживання такі похідні одиниці, як:

- 1 Терабайт (Тбайт) = 1024 Гбайт = 2^{40} байт,
- 1 Петабайт (Пбайт) = 1024 Тбайт = 2^{50} байт.

Література: [осн. 1, 2].

Завдання на СРС: Вивчення матеріалу лекції. Ієрархічні структури даних.
Література: [осн. 1, 2].

Лекція 2

Тема: Система збереження інформації в комп'ютері. Типи і види файлів. Робота з файлами. Макетування документів в Word

Основою системи збереження інформації в комп'ютері є спеціальні пристрої – диски. На комп'ютері їх може бути декілька. Кожен диск має назву (ім'я диску). Ім'я диску складається з латинської букви і двокрапки. Наприклад: **C:** або **A:**. Будь-який диск (незалежно від його фізичної суті) для користувача є сховищем інформації, до якого він звертається за ім'ям.

Перелік дисків, встановлених на комп'ютері, можна побачити, якщо відкрити вікно [Мой компьютер \(My computer\)](#),

Інформація в комп'ютері зберігається у вигляді файлів. *Файл* – це сукупність даних, яка зберігається на дисках та має ім'я.

Ім'я файлу складається з двох частин. Перша частина – це безпосереднє *ім'я файлу*. Друга – *розширення імені*, яке вказує на *тип* файлу. Ім'я файлу може бути довжиною до 256 символів і складатися з довільної послідовності латинських літер, літер кирилиці, цифр, знаків підкреслення, тире, пробілу та таке інше. Існують символи, які не можна включати до імені файлу.

Заборонені для імені файлу символи: * : " ? / \ < >

Розширення імені може бути довжиною до 4 символів (зазвичай – 3) і складається з латинських літер. Розширення імені - це абревіатура, яка вказує на тип файлу. Тип файлу, в свою чергу, вказує операційній системі, яким чином треба опрацьовувати цей файл. Основні типи файлів наведені в табл. 1.

Таблиця 1 – Основні типи файлів

Розширення імені файлу (тип файлу)	Призначення
1	2
.txt	Текстовий файл. Відкривається програмою Блокнот (Notepad)
.jpg, .jpeg, .bmp	Файли малюнків
.doc, .docx	Текстовий файл. Відкривається програмою Microsoft Word
.xls, .xlsx	Файл електронних таблиць. Відкривається програмою Microsoft Excel
.ppt	Файл презентації. Відкривається програмою Microsoft PowerPoint
.exe, .com	Файли програм. Запускаються на виконання
.zip, rar	Архівні файли. Зберігають інші файли у стислому стані
.html	Файл Web-сторінки. Переглядається у вікні Web-оглядача

Система збереження інформації передбачає розміщення папок на кожному диску. В папках можуть розташовуватись файли, а також інші папки. Розміщення одних папок усередині інших утворюють ієрархію («дерево») системи збереження інформації на комп'ютері.

Папки також мають ім'я, вимоги до яких співпадають з вимогами імені файлу.

Для того, щоб переглянути структуру («дерево») збереження інформації зручно використовувати на робочому столі комп'ютеру провідник МОЙ КОМП'ЮТЕР. Доступ до цієї програми можна знайти знизу ліворуч через ПУСК.

1 Види файлів

Файли можна умовно розділити на три види: 1. файли даних (файли або документи); 2. програмні файли; 3. системні файли.

1.1 Робота з файлами

Розглянемо операції з файлами та папками.

Не рекомендуємо виконувати операції з файлами та папками, які відносяться до операційної системи. Це може призвести до перебою в роботі операційної системи.

Робота з файлами складається з таких операцій:

1. створення файлу або папки;
2. перейменування файлу або папки;
3. копіювання файлу або папки;
4. переміщення файлу або папки;
5. видалення файлу або папки.

Всі операції з файлами виконуються за загальним алгоритмом. Деякі з операцій обробки файлів і папок можна виконати відразу для декількох об'єктів.

Алгоритм дій роботи з файлами та папками:

1. Виберіть папку, що містить потрібну папку або файл;
2. Виділіть папку або файл, що підлягає обробці;
3. Виберіть необхідну операцію обробки.

Розглянемо як відбувається виділення папок і файлів.

Робота з файлами виконується у вікні [Мой компьютер \(My Computer\)](#). Існують й інші програмні засоби для виконання операцій над файлами й папками, однак у даному курсі вони не розглядаються.

Для виділення *одного об'єкту* (папки або файлу) треба натиснути на ньому лівою клавішею миші.

Для виділення *декількох об'єктів*, які знаходяться *поруч один з одним*.

1. Виділіть перший об'єкт.
2. Натисніть клавішу **<Shift>** й, утримуючи її натиснутою, натисніть на ліву клавішу миші на останньому об'єкті групи, яка виділяється. Всі об'єкти, розташовані між цими двома, будуть виділені.

Для виділення *декількох об'єктів*, які знаходяться *не поруч один з одним*.

1. Виділіть перший об'єкт.
2. Натисніть клавішу **<Ctrl>**. Утримуючи її натиснутою, натисніть на ліву клавішу миші на наступному об'єкті. Обидва об'єкти виявляться виділеними.
3. Продовжуючи втримувати клавішу **<Ctrl>**, натисніть на ліву клавішу миші мишею на інших об'єктах, що підлягають виділенню.

Для виділення *всіх об'єктів* у вікні.

Виконаєте команду [Правка \(Edit\) — Выделить все \(Select All\)](#) або натисніть **<Ctrl>+<F>**. Незалежно від виділень, що виконувалися раніше, ця команда виділить всі об'єкти в даному вікні.

Для скасування виділення.

1. Щоб скасувати виділення окремого об'єкта в групі, натисніть клавішу **<Ctrl>** й, утримуючи її натиснутою, натисніть на ліву клавішу миші на відповідному об'єкті.
2. Щоб скасувати виділення всіх об'єктів, натисніть на ліву клавішу миші в будь-якому пустому місці у вікні.

Розглянемо як відбувається створення нової папки.

1. Відкрийте вікно пристрою або папки, де повинна бути створена нова папка.
2. Виконайте команду меню [Файл \(File\) — Создать \(Create\) — Папка \(Folder\)](#). З'являється значок нової папки з тимчасовим ім'ям [Новая папка \(New Folder\)](#). Це ім'я відразу можна замінити на інше.
3. Введіть власне ім'я папки. При введенні першого символу нового імені папки тимчасове ім'я видаляється.
4. Натисніть **<Enter>**. Процедура створення нової папки завершена. Пізніше папку можна перейменувати.

Команду [Создать \(Create\) — Папка \(Folder\)](#) можна викликати також за допомогою контекстного меню. За допомогою контекстного меню нову папку можна створити безпосередньо на Робочому столі.

Розглянемо як відбувається перейменування папок або файлів.

1. Виділіть папку або файл, які хочете перейменувати.
2. Виконаєте команду меню [Файл \(File\) — Переименовать \(Rename\)](#). Ім'я папки або файлу буде виділено.
3. Уведіть нове ім'я папки або файлу або відредагуйте існуюче.
4. Натисніть **<Enter>**.

Для перемикання мови клавіатури в режимі перейменування можна користуватися клавішами перемикання клавіатури **<Ctrl>+<Shift>** або **<Alt>+<Shift>** (у лівій частині клавіатури), а не індикатором панелі завдань

Інші способи:

1. Викличте контекстне меню на обраній папці або файлі. У контекстному меню виберіть команду [Переименовать \(Rename\)](#).

Або:

2. Ще раз натисніть на ліву клавішу миші на імені виділеної папки або файлу.

При цьому не слід виконувати друге клацання слідом за першим, щоб система не сприйняла їх як подвійне клацання: подвійним клацанням система відкриває файл або папку.

Розглянемо як відбувається копіювання папок або файлів.

Папки й файли можна перемістити або скопіювати за допомогою буфера обміну. У цьому випадку алгоритм буде виглядати в такий спосіб:

1. Виділіть потрібні папки або файли.
2. Виконайте команду меню [Правка \(Edit\) — Копировать \(Copy\)](#).
3. Відкрийте папку, у яку хочете скопіювати обрані об'єкти.
4. Виконайте команду меню [Правка \(Edit\) — Вставить \(Paste\)](#).

Замість команд меню [Правка \(Edit\)](#) для роботи з буфером можна використати аналогічні команди контекстного меню.

У випадку, якщо на екрані присутні два вікна папок - то вікно, ЗВІДКИ копіюється папка або файл і те вікно, КУДИ *копіюється папка або файл*, зручно користуватися мишею. У цьому випадку послідовність дій буде виглядати в такий спосіб:

1. Виділити потрібний файл або папку.
2. Нажавши й не відпускаючи праву клавішу миші, перетягнути обраний об'єкт із одного вікна (папки) в інше вікно.
3. Після відпускання кнопки миші на екрані з'являється контекстне меню. Виберіть у ньому команду [Копировать \(Copy\)](#).

Якщо папки або файли повинні бути скопійовані на Робочий стіл, виділіть їх у вікні й перетягніть у вільну зону на робочому столі при натиснутій правій кнопці миші. Далі дійте у відповідність із попереднім алгоритмом.

Розглянемо як відбувається переміщення папок або файлів.

При використанні буфера обміну операція переміщення відрізняється від операції копіювання тільки другим кроком алгоритму.

1. Виділіть потрібну папку або файл.
2. Виконайте команду меню [Правка \(Edit\) — Вырезать \(Cut\)](#).
3. Відкрийте папку, у яку хочете перенести виділений об'єкт.
4. Виконайте команду [Правка \(Edit\) — Вставить \(Paste\)](#).

Так як і операція копіювання, операція переміщення може бути виконана різними способами.

Розглянемо як відбувається видалення папки/файлу.

Непотрібні папки й файли можна видалити. При видаленні папки будуть вилучені всі об'єкти, збережені в ній.

1. Виділіть папку або файл, які необхідно видалити.
2. Виконайте команду [Файл \(File\) — Удалить \(Delete\)](#).

Або:

Натисніть клавішу *<Delete>* на клавіатурі.

3. За замовчуванням на екрані з'являється запит на підтвердження видалення

Для підтвердження видалення потрібно натиснути на кнопці [Да \(Yes\)](#). Натискання на кнопці [Нет \(No\)](#) скасовує операцію видалення.

Видалення приводить до переміщення вилучених папок або файлів у папку [Корзина \(Recycle Bin\)](#). З корзини вилучені папки й файли можуть бути видалені або відновлені пізніше.

Однак папка [Корзина \(Recycle Bin\)](#) може бути настроєна таким чином, що файли будуть автоматично видалятися без можливості відновлення. Тому операція видалення може привести до остаточної втрати інформації. Тому треба бути обережним, виконуючі операцію видалення.

Література: [осн. 1, 2].

Завдання на СРС: Вивчення матеріалу лекції. Робота з файлами. Підбір матеріалів попереднього вивчення про інформатику. **Література:** [осн. 1, 2].

Лекція 3

Тема: Кодування чисел двійковим кодом. Властивості двійкової системи кодування даних Різноманітність кодування текстових даних, графічних даних, звукових даних. Кількість інформації

Як було сказано вище, інформаційний обмін здійснюється у вигляді кодованих сигналів. На сучасному рівні може бути закодовано практично будь-яке поняття або характеристика фізичного об'єкта. Однак треба зазначити (і це головне), що кодування в комп'ютерній техніці прийнято цифрове. Тобто визначеному поняттю ставиться у відповідність певне число. Так, наприклад, кодуються букви і символи алфавіту (кожному символу алфавіту у відповідність ставиться деяке ціле число, наприклад, порядковий номер у списку букв і символів).

Виникає питання про кодування власне чисел. Але, як про це говорилося кодування чисел якраз і не викликає технічних складнощів в електроніці. Для цього можна використовувати двійкову систему числення і принцип електричного кодування (тобто є струм в ланцюзі чи ні).

Кодування за допомогою двійкової системи числення отримало розвиток в комп'ютерній техніці.

Початок двійкової системи числення належить Лейбніцем.

Приклад з оленями і чукчами. Кожній фізичній об'єкту поставлений у відповідність також фізичному об'єкту: олень - палець.

У римлян існувала десяткова система числення, але вкрай незручна.

Арабська - десяткова система числення поклала початок поданням числа, як абстрактного поняття.

Домовимося позначати фарбою кожен знову з'явився в комірці кульку. Якщо кульки немає, то ми її і не «фарбуємо». Таким чином:

З'явилася кулька – 1

З'явилася друга кулька - їх вже дві; і маємо право використовувати тільки два символи 0 і 1.

Для двох кульок я їх використовую «2» = 10_2

Як порахувати трійку «двійками»? Для початку потрібно подивитися, як змінюються розряди.

Таблиця 1- Умовні позначення кодування у двійковій системі через кульки

	○	○ ○	○ ○ ○	○ ○ ○ ○ ○ ○ ○ ○	○ ○ ○ ○ ○	○ ○ ○ ○ ○ ○	○ ○ ○ ○ ○ ○ ○ ○ ○	○ ○ ○ ○ ○ ○ ○ ○ ○ ○	
	○	● ○ ○	● ○ ○ ○	● ○ ○ ○ ○	● ○ ○ ○ ○ ○	● ○ ○ ○ ○ ○ ○	● ○ ○ ○ ○ ● ○ ○ ○		
0	1	10	11	100	101	110	111	1000	1001

Власне кодування цілих чисел у двійковій системі досить просто. Воно може бути представлене схематично наступним чином. *Правило кодування наступне:*

Двійкове число, яке виражає те ж кількісне поняття, виходить як сукупність залишків від послідовного ділення на два, записана справа наліво.

Приклад, як можна представити у двійковій системі ціле число 37:

$$37 : 2 = 18 + 1$$

$$18 : 2 = 9 + 0$$

$$9 : 2 = 4 + 1$$

$$4 : 2 = 2 + 0$$

$$2 : 2 = 1 + 0$$

$37_{10} = 100101_2$ –цей запис означає, що десяткове число 37 відповідає двійковому числу 100101

Якщо представити перші 16 чисел (від 0 до 15) в двійковому вигляді, то вони будуть мати вигляд:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Десяткове															
0	1	10	11	100	101	110	111	1000	1001	1010	1011	1100	1101	1110	1111
Двійкове															

1 біт = два числа

2 біта = два числа

3 біта = чотири числа
Не всі поєднання використовуємо

4 біта = вісім чисел

Для пояснення двійкового запису проведемо аналогію зі звичним нам десятковим записом чисел і зміною розряду 10:

1	2	3	4	5	6	7	8	9	10
$10^0 = 1$									10^1
Значущі цифри десяткового розряду									
Зміна розряду				Зміна розряду					
10	20	30	40	50	60	70	80	90	100
$10^1 = 10$									$10^2 =$
100	Зміна розряду							Зміна розряду	

Легко помітити на прикладі зміни розряду десяткової системи числення, що на зміну розряду вказує додавання нуля до попередньої старшої цифри розряду:

1 10 100 1000 і т.д.

Відповідно, наступний розряд визначається ступенем числа 10, тобто основи даної (десяткової системи числення):

10^0 10^1 10^2 10^3 і т.д.

Порівнюючи десяткові ряди чисел з рядом чисел двійкової системи можна отримати повну аналогію зміни розрядів:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Десяткове															
0	1	10	11	100	101	110	111	1000	1001	1010	1011	1100	1101	1110	1111
Двійкове															

Зміна розряду $2^2 = 4$

Зміна розряду $2^3 = 8$

Зміна розряду: $2^1 = 2$

Зміна розряду: $2^0 = 1$ (в двійковій системі - дві значущі цифри «0» і «1»; далі міняється розряд

Зі сказаного легко зрозуміти, як може бути записано будь-яке число у вигляді суми двійок у відповідній мірі:

Наприклад, те ж число 37 може бути представлено, як

$$37 = 2^0 + 2^2 + 2^5 = 1 + 4 + 32$$

(1) Питання рейтингове: чи можна уявити «37» у вигляді іншого сполучення ступенів «2»

(2) Чи може двійковий код числа закінчуватися «0»

Для зворотного декодування числа з двійкової системи в десяткову ($37_{10} = 100101_2$) існує таке правило:

а) необхідно під кожною цифрою двійкового числа підписати двійку в ступені, починаючи від нульової справа наліво:

1	0	0	1	0	1
2^5	2^4	2^3	2^2	2^1	2^0

б) залишити числа тільки навпроти одиниць в двійковому записі числа

1	0	0	1	0	1
2^5	2^4	2^3	2^2	2^1	2^0
↓			↓		↓
2^5	0	0	2^2	0	2^0

в) виконати додавання чисел:

$$32 + 0 + 0 + 4 + 0 + 1 = 37$$

Кодування даних в обчислювальній техніці

Для кодування інформації використовується двійковий код. Розглянемо схему кодування і передачі інформації машиною і людиною. Нехай необхідно вирішити обчислювальну задачу:

$$2 \times 3 = ?$$

Задача кодування цієї задачі словами відома і зрозуміла кожному: «Скільки буде: два помножити на три?».

Для передачі цієї інформації приймачу ми закодували словами математичні дії і дані. Приймач, одержавши інформацію, порівнює слова з наявним у нього в пам'яті

декодером, записує інформацію, складає алгоритм, а потім обробляє для обчислень. Для людини ця операція дуже проста в розумінні кодування, складання алгоритму і вирішення. Комп'ютер не може сам закодувати інформацію, а тим більше скласти алгоритм.

У чому ж переваги комп'ютера?

Таблиця 2- Порівняння сприйняття інформації людиною і комп'ютером

Завдання	Людина	Комп'ютер
„Скільки буде два помножити на три”	Кодування словами: букви, звуки. Поняття, закріплені навичками і навчанням	Кодування цифрами. Програми для декодування і відображення даних
	Сам складає алгоритм	Потрібно завдання алгоритму зовнішнім джерелом
	Сам навчається кодуванню, складання алгоритму і т.д.	Вимагає внесення програм для обробки і обчислень
	Повільно обчислює	Обчислює дуже швидко

Кодування текстових даних

У розпорядженні програмістів, які складають програми для кодування і декодування інформації комп'ютером, є тільки дві цифри «0» і «1». Ці цифри необхідно якимось чином пристосувати для кодування різної інформації. Зокрема, текстова інформація може бути закодована, якщо кожній букві і символу алфавіту присвоїти число. Підраховано, що для кодування всіх літер і символів російського або англійського алфавіту достатньо 256 комбінацій цифр. Якщо зіставити це число з двійковим кодом, то можна отримати:

$$256 = 2^m$$

де m - розряд двійкового коду, тобто скільки цифр (бітів) повинні скласти розрядність коду.

Звідси розрядність коду легко визначається, як

$$m = \frac{\ln 256}{\ln 2} = \frac{\ln(2^8)}{\ln 2} = \frac{8 \ln 2}{\ln 2} = 8 ,$$

тобто 8-ми розрядний код.

Таким чином можна закодувати будь-яку букву алфавіту. Наприклад, в англійському кодуванні за допомогою так званого ASCII (American National Standart Code for Information Interchange) літера «А» має порядковий номер 65. Як буде виглядати її двійкове відображення в 8-ми розрядному двійковому коді - знають програмісти, які записують базові символи ASCII-системи. Наше завдання зрозуміти принцип кодування, а не конкретне зображення того чи іншого символу.

Система UNICODE заснована на 16 розрядному кодуванні символів, тому в цій системі можна закодувати 65536 різних символів, що дозволяє закодувати в одній таблиці більшість мов всієї планети. Довгий час система UNICODE не реалізовувалася через обмеженість можливостей обчислювальної техніки.

Технічний устрій дисплея такий, що кожна невелика діляночка екрану складається з сукупності трьох точок, кольори яких в сумі передають природний колір предмета в деякому масштабі. Екран має свій розмір, виражений в умовних розмірних одиницях, т. зв. пікселях, і тому може бути зіставлений з декартової системою координат. У відповідності з цим, кожна точка екрану має свої координати, які можуть бути також закодовані двійковим кодом. У свою чергу кожна точка екрану може бути представлена з різною яскравістю. Для кожної точки прийнято 256 градацій сірого кольору (від повністю темною до повністю відкритої для внутрішнього освітлення монітора).

Для кодування кольорових графічних зображень застосовується принцип декомпозиції будь-якого кольору на основні складові: червоний (Red, R), зелений (Green, G), синій (Blue, B). На практиці будь-який колір можна представити комбінацією цих трьох кольорів. Система кодування називається RGB. Якщо для кодування яскравості кожної з основних складових однієї точки використовувати 24 розряду, то можна отримати

$$N = 2^{24} = 1.678 \cdot 10^7 \text{ різних кольорів,}$$

що близько до чутливості людського ока.

В даний час використовується також додаткові кольори до основних:

(Cyan, C) (Magenta, M) (Yellow, Y). Для представлення точки в цьому випадку необхідно вже 32 розрядний код.

Файли і файлова структура

В інформатиці існує двійкова система представлення даних. Найменшою одиницею такого подання є біт (двійковий розряд - «1» або «0»).

Сукупність двійкових розрядів утворює так званий бітовий малюнок. Для впорядкування бітового малюнка кодована інформація розбивається на групи з восьми бітів. Вісім біт утворюють байт.

Таким чином запис чисел у вигляді байтового представлення (бітового малюнка) має вигляд (Табл.2).

Таблиця 3 - Запис десяткових і двійкових чисел у вигляді байтового представлення

Десяткове число	Двійкове число	Байт
1	1	0000 0001
2	10	0000 0010
....
255	1111 1111	1111 1111

На сьогоднішній день крім байтової структури використовуються також 16-розрядне, 24-розрядне ... 64-розрядне кодування. ($2^{64} = 1.845 \times 10^{19}$).

Повертаючись до прикладу з відомостями про студентів для деканату, задамо собі питання, однакове чи час необхідно приймачу інформації (деканату) для того, щоб обробити отримані дані і зробити адекватний висновок. Ясно, що різний час знадобиться для перевірки наявності студентів на знімку і читання списку тільки відсутніх на лекції.

Для оцінки кількості інформації, що міститься в повідомленні, використовують формулу Хартлі (1928 р.). Хартлі – американський інженер.

Кількість інформації в повідомленні

$$M = \log_2 N,$$

де N – наперед задана множина (кількість, число) рівноймовірних повідомлень, одержуваних приймачем інформації.

Так як в математиці є обчислені (табличні) значення функцій десяткового і натурального логарифмів, то логарифм за основою «2» може бути перетворений на дріб по їх основі:

$$\log_2(N) = M$$

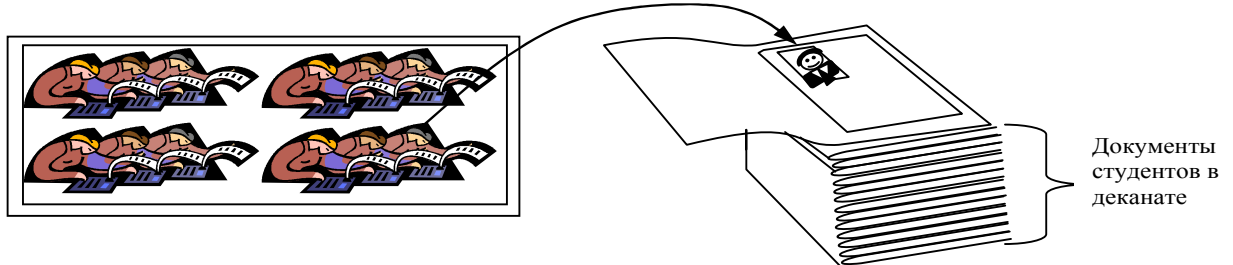
$$N = 2^M$$

$$\ln(N) = M \ln 2 = \log_2(N) \ln 2$$

$$\log_2(N) = \frac{\ln(N)}{\ln 2}$$

Таким чином, деканат отримає різну кількість інформації в різних повідомленнях. Для отримання висновку деканату необхідно виконати наступне декодування даних:

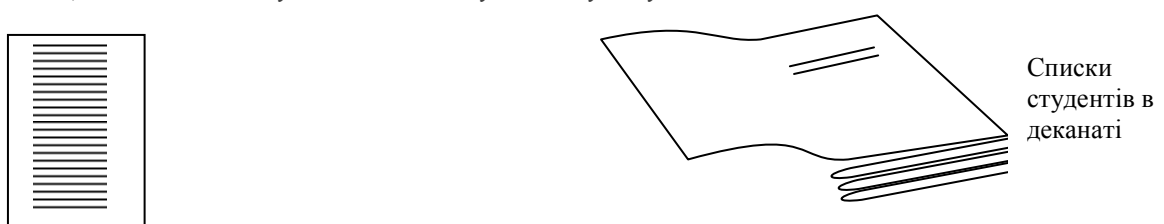
а) повідомлення у вигляді фотографії потоку



Для того, щоб з'ясувати які студенти відсутні деканат повинен порівняти зображення студентів на фотографії і в документах. Для цього потрібно переглянути 100 фотографій і 100 прізвищ (стільки студентів на потоці). Тобто фотографія потоку несе кількість інформації

$$M = \log_2(200) = \frac{\ln(200)}{\ln 2} \approx 7,644 \text{ одиниць інформації}$$

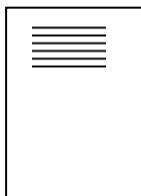
б) повідомлення у вигляді списку потоку студентів



Для того, щоб з'ясувати які студенти присутні, деканат повинен порівняти списки груп студентів. Для цього потрібно переглянути 4 списки груп із 100 прізвищами (стільки студентів на потоці). Тобто список потоку несе кількість інформації

$$M = \log_2(100) = \frac{\ln(100)}{\ln 2} \approx 6,644 \text{ одиниць інформації}$$

в) повідомлення у вигляді списку відсутніх студентів



Для того, щоб з'ясувати, які студенти відсутні, деканат взагалі не повинен виконувати ніякої додаткової роботи з ідентифікації студентів. Тобто список відсутніх (відсутні 5 студентів) несе кількість інформації

$$M = \log_2(5) = \frac{\ln(5)}{\ln 2} \approx 2,322 \text{ одиниць інформації}$$

Ще один приклад. Припустимо, потрібно вгадати одне ціле число з набору чисел від одиниці до десяти тисяч. *За формулою Хартлі* можна обчислити, яка кількість інформації для цього потрібна:

$$I = \log_2 10000 \approx 13,288 \quad (\ln(10000) / \ln(2) = 13,288),$$

тобто що нам повинні повідомити про число, щоб ми змогли його назвати, скільки «наведень»?

Тобто повідомлення про вірно вгадане число містить кількість інформації, приблизно рівну 13,288 одиниць інформації. Як бачимо, кількісна оцінка кількості інформації, недостатньо наочна, вона не говорить нам про те, багато це чи мало для отримання відповіді. Зрозуміло, що для вгадування одного числа з набору в 10000 необхідно досить велика кількість інформації, в порівнянні з прикладом з фотографіями. При цьому кількісна оцінка збільшилася, наприклад, у порівнянні з наведеним вище прикладом незначно (порівняйте 6,644 і 13,288).

Як можна собі уявити необхідну кількість інформації для вгадування числа? Можна це виконати за наступним алгоритмом:

Перша ітерація (наближення): Необхідно отримати інформацію про те, в якій області знаходиться шукане число А. Необхідно отримати від джерела інформації вказівку:

$$A < 5000$$

або

$$5000 < A < 10000$$

Друга ітерація: Якщо $A > 5000$, то тоді

Необхідно отримати від джерела інформації вказівку:

$$7500 > A > 5000$$

або

$$7500 < A < 10000$$

і т. д., поки не отримаємо інтервал в три цілих числа.

Спробуйте оцінити кількість ітерацій для визначення шуканого числа і порівняйте з отриманим параметром «Кількість інформації». Необхідно відзначити, що використаний прийом з пошуку числа має в математиці досить суворе обґрунтування і використовується при вирішенні алгебраїчних рівнянь.

Якщо події носять нерівновирогідний характер, то в цьому випадку можна використовувати формулу, запропоновану американським ученим Клодом Шенноном.

Кількість інформації за формулою К. Шеннона:

$$M = - (p_1 \log_2 p_1 + p_2 \log_2 p_2 + \dots + p_N \log_2 p_N),$$

де p_i - імовірність того, що саме i -те повідомлення виділено в наборі з N повідомлень.

Якщо ймовірності p_1, \dots, p_N рівні, то кожна з них дорівнює $1/N$, і формула Шеннона перетворюється на формулу Хартлі.

$$\begin{aligned} -(p_1 \log_2 p_1 + p_2 \log_2 p_2 + \dots + p_N \log_2 p_N) &= -p_i [\log_2 (p_1 \cdot p_2 \cdot \dots \cdot p_N)] = -p_i \log_2 (p_i^N) = \\ &= -\frac{1}{N} N \log_2 (p_i) = -\log_2 \left(\frac{1}{N} \right) = \log_2 N \end{aligned}$$

Наступним етапом у характеристиці кількості інформації є введення одиниці кількості інформації. Для пояснення цього етапу необхідно зробити історичний огляд обчислювальної техніки. Більш докладно про це будемо говорити пізніше. Зараз вкажемо лише на те, що в перших прообразах сучасних комп'ютерів (т. зв. електронних обчислювальних машинах, або ЕОМ) використовували принцип розпізнавання інформації по наявності або відсутності електричних сигналів. Тобто якщо струм протікає через ланцюг - сигнал є, якщо немає струму - сигнал відсутній. Тоді для інформаційного обміну в нашому розпорядженні з'являється дві умови:

- Сигнал у ланцюзі є - позначимо такий стан «1»;
- Сигнал відсутній - позначимо такий стан «0».

Спробуємо виконати наведений вище алгоритм з пошуку числа символно (Табл.4).

Таблиця 4 - Алгоритм з пошуку числа символно

Ітерація	Математичний зміст ітерації	Кількісна оцінка змісту ітерації
1	$5000 < A < 10000$	«Так» = 1
	$0 < A < 5000$	«Ні» = 0
2	$7500 < A < 10000$	«Ні» = 0
	$5000 < A < 7500$	«Так» = 1
3	$5000 < A < 6250$	«Так» = 1
	$6250 < A < 7500$	«Ні» = 0
4	$5625 < A < 6250$	«Ні» = 0
	$5000 < A < 5625$	«Так» = 1
5	$5312 < A < 5625$	«Ні» = 0
	$5000 < A < 5312$	«Так» = 1
6	$5156 < A < 5312$	«Ні» = 0
	$5000 < A < 5156$	«Так» = 1
7	$5078 < A < 5156$	«Ні» = 0
	$5000 < A < 5078$	«Так» = 1
8	$5039 < A < 5078$	«Ні» = 0
	$5000 < A < 5039$	«Так» = 1
9	$5019 < A < 5039$	«Ні» = 0
	$5000 < A < 5019$	«Так» = 1
10	$5009 < A < 5019$	«Ні» = 0
	$5000 < A < 5009$	«Так» = 1
11	$5004 < A < 5009$	«Ні» = 0
	$5000 < A < 5004$	«Так» = 1
12	$5002 < A < 5004$	«Ні» = 0
	$5000 < A < 5002$	«Так» = 1
13	$5001 < A < 5002$	«Ні» = 0
	$5000 < A < 5001$	«Так» = 1

і т.п., поки не отримаємо шуканий результат. Загалом має бути не менш 13 ітерацій.

В даному описі принципу пошуку заданого числа для нас важливо відзначити, що висновок ми можемо отримати, обробляючи інформацію тільки з урахуванням двох чисел «1» і «0». А як було сказано вище щодо ЕОМ це якраз відповідає принципу електричної фіксації сигналів. Тобто для ЕОМ можливе використання тільки двох цифр «1» і «0». Така система числення відома в математиці і називається двійковою. У двійковій системі

числення «0» і «1» називаються двійковими цифрами. Ці ж цифри і ця ж назва покладені в основу кількісного виміру інформації.

Звідси і основа логарифма 2.

Література: [осн. 1, 2].

Завдання на СРС: Вивчення матеріалу лекції. Розв'язання задачі за допомогою формули кількості інформації, розв'язання задач за темою: приклади кодування та декодування. Загальні відомості про інтерактивні методи у інформатиці **Література:** [осн. 1, 2].


Лекція 4

Тема: Особливості автоматичних розрахунків на ПЕОМ

Принцип роботи комп'ютера

Прототипом комп'ютера можна вважати, наприклад, механічні годинники. Для чого призначені годинники? Для того, щоб показувати результат відліку часу. Для цього механізм годинника із заданою частотою переміщає показчик часу (по суті, показчик чисел - секунд, хвилин, годин), а власник годин (користувач) фіксує необхідний йому відлік. Так само принципово функціонує і комп'ютер.

У комп'ютері є особливий пристрій - генератор, який виробляє електричний сигнал

певної частоти. Сам по собі сигнал має вигляд, показаний на малюнку. →  Електроніки називають такий сигнал «пила». Як правило, сигнали такої конфігурації називаються несучими і здатні передавати інформацію, якщо на «пилку» накласти електричний сигнал іншої конфігурації (звук, колір) або оцифрувати кожен пік «пили» (числа, цифровий код). У разі передачі звуку на несучу частоту накладається електричний сигнал, аналогічний акустичним коливанням, передавальним звук. Такий принцип передачі називається аналоговим. Для такої передачі не потрібно великої несучої частоти. Принципово передача інформації «пилою» у вигляді електромагнітних хвиль (радіохвилі) відрізняється несучою частотою генератора. Цифрова передача вимагає дуже великих частот генератора, так як кожен пік несе на собі цифру (0 або 1). Сучасні комп'ютери мають частоти генераторів більш 10^9 Гц, т.е. більше 1 мільярда коливань в секунду.

Повернемося до прототипу комп'ютера - механічного годиннику. Годинник через рівні проміжки часу генерує переміщення стрілки годинника, які можна вважати деякими сигналами. При русі стрілки сигнали підсумовуються, тобто виконуються якісь математичні дії. Всі дії годин виконуються автоматично по заданому закону (алгоритму): повний обхід циферблата відповідає певному числу сигналів (секунд, хвилин, годин).

Таким чином, годинник - це пристрій, який автоматично виконує підсумовування числа даних (секунд, хвилин, годин) і реєстрацію результатів підсумовування.

Принципово, те ж саме робить електронно-обчислювальна машина (ЕОМ).

Для поглиблення розуміння роботи комп'ютера необхідно розібратися в двійковій арифметиці, так як двійкова система числення є однією з математичних основ дії комп'ютера.

Складання двійкових чисел

Спосіб додавання стовпчиком такий же, як і для десятичного числа. Тобто, складання виконується порозрядно, починаючи з молодшої цифри. Якщо при додаванні двох цифр виходить СУМА більше дев'яти, то записується цифра = СУМА-10, а ЦІЛА ЧАСТИНА (СУМА / 10), додається в старшому розряді. (Складіть пару чисел стовпчиком

згадайте як це робиться). Так і з двійковим числом. Складаємо порозрядно, починаючи з молодшої цифри. Якщо виходить більше 1, то записується 1 і 1 додається до старшого розряду.

Виконаємо приклад: $10011 + 10001$.

	1	0	0	1	1
	1	0	0	0	1
1	0	0	1	0	0

Перший розряд: $1+1=2$. Записуємо 0, і 1 додається до старшого розряду.

Другий розряд: $1+0+1$ (Запам'ятована одиниця) $=2$. Записуємо 0, і 1 додається до старшого розряду.

Третій розряд: $0+0+1$ (Запам'ятована одиниця) $=1$. Записуємо 1.

Четвертий розряд: $0+0=0$. Записуємо 0.

П'ятий розряд: $1+1=2$. Записуємо 0, і додаємо до шостого розряду 1.

Переведемо всі три числа в десяткову систему і перевіримо правильність складання.

$$10011 = 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 16 + 2 + 1 = 19$$

$$10001 = 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 16 + 1 = 17$$

$$100100 = 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 32 + 4 = 36$$

$$17 + 19 = 36 - \text{вірна рівність.}$$

Приклади для самостійного розв'язання:

а) $11001 + 101 =$

б) $11001 + 11001 =$

с) $1001 + 111 =$

д) $10011 + 101 =$

е) $11011 + 1111 =$

д) $11111 + 10011 =$

Віднімання двійкових чисел

Віднімати числа будемо також стовпчиком і загальне правило те ж, що і для десяткових чисел, віднімання виконується порозрядно і якщо в розряді не вистачає одиниці, то вона займається в старшому. Розв'яжемо наступний приклад:

	1	1	0	1
-		1	1	0
		1	1	1

Перший розряд. $1 - 0 = 1$. Записуємо 1.

Другий розряд $0 - 1$. Не вистачає одиниці. Займаємо її в старшому розряді. Одиниця зі старшого розряду переходить в молодший, як дві одиниці (тому що старший розряд представляється двійкою більшого ступеня) $2 - 1 = 1$. Записуємо 1.

Третій розряд. Одиницю цього розряду ми займали, тому зараз в розряді 0 і є необхідність зайняти одиницю старшого розряду. $2 - 1 = 1$. Записуємо 1.

Перевіримо результат в десятковій системі
 $1101 - 110 = 13 - 6 = 7$ (111) – Вірна рівність.

Ще один спосіб виконання віднімання пов'язаний з поняттям додаткового коду, який дозволяє звести віднімання до додавання. Правило віднімання через додатковий код стверджує, що віднімання можна замінити на додавання, якщо від'ємник замінити на число в додатковому коді. Беремо число, замінюємо нулі на одиниці, одиниці навпаки замінюємо на нулі і до молодшого розряду додаємо одиницю. Наприклад, 10010, в додатковому коді буде 011011.

Приклад: $34 - 22 = 12$

Запишемо цей приклад в двійковому вигляді.

$$100010 - 10110 = 1100$$

Додатковий код числа 10110 буде таким:

$$01001 + 00001 = 01010.$$

Тоді вихідний приклад можна замінити складанням так:

$$100010 + 01010 = 101100$$

Далі необхідно відкинути одну одиницю в старшому розряді. Якщо це зробити, то отримаємо 001100. Відкинемо незначущі нулі і отримаємо 1100, тобто приклад розв'язаний правильно.

Завдання на СРС: Виконайте віднімання звичайним способом і в двійковому коді, перевіривши попередньо десяткові числа в двійкові:

- а) $456 - 112$
- б) $234 - 12$
- в) $345 - 232$
- г) $456 - 78$
- д) $567 - 109$
- е) $67 - 45$

Множення в двійковій системі числення.

Для початку розглянемо наступний цікавий факт. Для того, щоб помножити двійкове число на 2 (десятькова двійка це 10 в двійковій системі) досить до множити числа зліва приписати один нуль.

Приклад. $10101 * 10 = 101010$

Перевірка.

$$10101 = 1*2^4 + 0*2^3 + 1*2^2 + 0*2^1 + 1*2^0 = 16 + 4 + 1 = 21$$

$$101010 = 1*2^5 + 0*2^4 + 1*2^3 + 0*2^2 + 1*2^1 + 0*2^0 = 32 + 8 + 2 = 42$$

$$21 * 2 = 42$$

Якщо ми згадаємо, що будь двійкове число розкладається за ступенями двійки, то стає ясно, що множення в двійковій системі числення зводиться до множення на 10 (тобто на десяткову 2), а стало бути, множення це ряд послідовних зрушень. Загальне правило таке: як і для десяткових чисел, множення двійкових виконується поразрядно. І для кожного розряду другого множника до першого множнику додається один нуль праворуч. Приклад (поки не стовпчиком): $1011 * 101$ Це множення можна звести до суми трьох порязрядних множень: $1011 * 1 + 1011 * 0 + 1011 * 100 = 1011 + 101100 = 110111$ У стовпчик це ж саме можна записати так:

		1	0	1	1
	*		1	0	1
		1	0	1	1
	0	0	0	0	
1	0	1	1		
1	1	0	1	1	1

Перевірка:

101 = 5 (десятькове)

1011 = 11 (десятькове)

110111 = 55 (десятькове)

5*11 = 55 правильне рівність

вирішите самостійно

а) 1101 * 1110 =

б) 1010 * 110 =

г) 1011 * 11 =

д) 101011 * 1101 =

е) 10010 * 1001 =

*Примітка: До речі таблиця множення в двійковій системі складається тільки з одного пункту $1 * 1 = 1$*

Ділення в двійковій системі числення

Ми вже розглянули три дії і думаю вже зрозуміло, що в загальному-то дії над двійковими числами мало відрізняються від дій над десятиковими числами. Різниця з'являється тільки в тому, що цифр дві а не десять, але це лише спрощує арифметичні операції. Так само йде справа і з поділом, але для кращого розуміння алгоритм ділення розберемо більш докладно. Нехай нам необхідно розділити два десятикових числа, наприклад 234 розділити на 7. Як ми це робимо.

2	3	4	7	

Ми виділяємо праворуч (від старшого розряду) таку кількість цифр, щоб вийшло число було якомога менше і в той же час більше дільника. 2 - менше дільника, отже, необхідне нам число 23. Потім ділимо отримане число на дільник із залишком. Отримуємо наступний результат:

	2	3	4	7	
-	2	1		3	
		2	4		

Описану операцію повторюємо до тих пір, поки отриманий залишок не опиниться менше дільника. Коли це станеться, число отримане під рисою, це приватне, а останній

залишок - це залишок операції. Так от операція ділення двійкового числа виконується точно також. спробуємо

Приклад: 10010111 / 101

1	0	0	1	0	1	1	1	1	0	1

Шукаємо число, від старшого розряду яке перше було б більше ніж дільник. Це чотирьох розрядне число 1001. Воно виділено жирним шрифтом. Тепер необхідно підібрати дільник виділеному числу. І тут ми знову виграємо в порівнянні в десятковою системою. Справа в тому, що підбирається дільник це обов'язково цифра, а цифри у нас тільки дві. Так як 1001 явно більше 101, то з дільником все зрозуміло це 1. Виконаємо крок операції.

	1	0	0	1	0	1	1	1	1	0	1
-		1	0	1					1		
		1	0	0							

Отже, залишок від виконаної операції 100. Це менше ніж 101, тому щоб виконати другий крок ділення, необхідно додати до 100 наступну цифру, це цифра 0. Тепер маємо наступне число:

	1	0	0	1	0	1	1	1	1	0	1
-		1	0	1					1		
		1	0	0	0						

1000 більше 101 тому на другому кроці ми знову допишемо в приватне цифру 1 і отримаємо наступний результат (для економії місця відразу опустимо наступну цифру).

	1	0	0	1	0	0	1	1	1	0	1
-		1	0	1					1	1	
		1	0	0	0						
	-		1	0	1						
				1	1	0					

Третій крок. Отримане число 110 більше 101, тому і на цьому кроці ми запишемо в приватне 1. Вийти так:

	1	0	0	1	0	0	1	1	1	0	1
-		1	0	1					1	1	1
		1	0	0	0						
	-		1	0	1						
				1	1	0					
			-	1	0	1					
							1	1			

Отримане число 11 менше 101, тому записуємо в приватне цифру 0 і опускаємо вниз наступну цифру. Виходить так:

	1	0	0	1	0	0	1	1		1	0	1		
-		1	0	1						1	1	1	0	
		1	0	0	0									
	-		1	0	1									
				1	1	0								
			-	1	0	1								
							1	1	1					

Отримане число більше 101, тому в приватне записуємо цифру 1 і знову виконуємо дії. Виходить така картина:

	1	0	0	1	0	0	1	1		1	0	1		
-		1	0	1						1	1	1	0	1
		1	0	0	0									
	-		1	0	1									
				1	1	0								
			-	1	0	1								
							1	1	1					
					-	1	0	1						
							1	0						

Отриманий залишок 10 менше 101, але у нас закінчилися цифри в подільному, тому 10 це остаточний залишок, а 1110 це шукане приватне. Перевіримо в десяткових числах

$$10010011 = 147$$

$$101 = 5$$

$$10 = 2$$

$$11101 = 29$$

	1	4	7	5	
-	1	0		2	9
		4	7		
	-	4	5		
			2		

Література: [осн. 1, 2].

Завдання на СРС: Вивчення матеріалу лекції. Розв'язання самостійно задач по темі: Додавання, віднімання, множення та ділення у двійковій системі числення.

Література: [осн. 1, 2].

Лекція 5

Тема: Поняття про операційну систему ПЕОМ. Програмні продукти ПЕОМ. Система Windows

Склад обчислювальної системи

Склад обчислювальної системи називається *конфігурацією* (Рис.1).

Розрізняють апаратну і програмну конфігурації. Із назв стає зрозуміло, що:

Апаратна конфігурація – це власне електронні блоки і системи, які є складовими комп'ютера, периферійні пристрої, які забезпечують електричний/електронний і логічний зв'язок між пристроями комп'ютера ;

Програмна конфігурація – це склад програмного забезпечення комп'ютера.

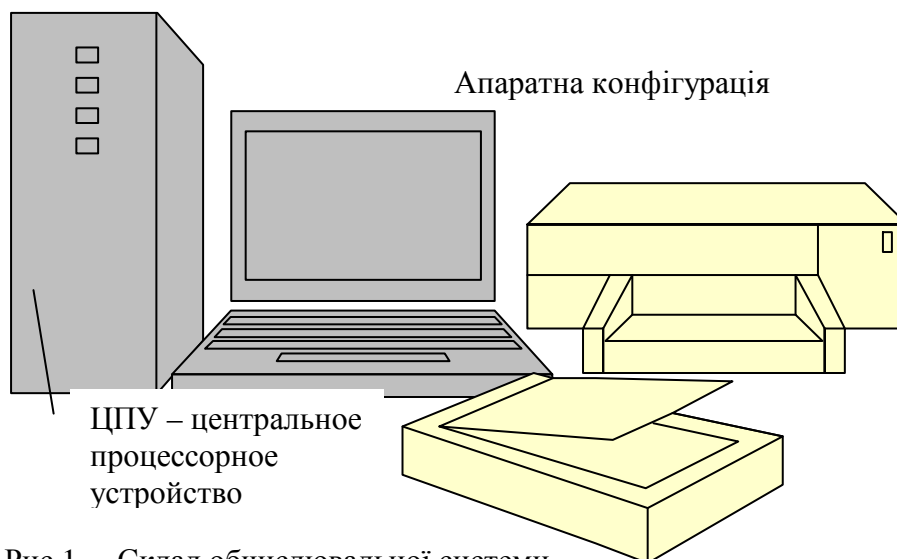


Рис.1 – Склад обчислювальної системи

Узгодження між пристроями забезпечуються за допомогою спеціальних пристроїв – *апаратних інтерфейсів*. *Інтерфейс* (англ. *inter face*) – дослівно «поміж обличчями»; діалог.

Інтерфейси призначені для передачі даних і забезпечують фізичне функціонування обчислювальної системи.

Засоби, які забезпечують функціонування обчислювальних процедур – *програмне забезпечення*. *Програми* – це впорядковані послідовності команд, які призводять до

певного результату (наприклад, обчислення і ін.). Програми працюють в нерозривному зв'язку одна з одною. Їх взаємодія забезпечується між *програмним інтерфейсом*.

Програмне забезпечення і апаратні інтерфейси працюють, як єдине ціле і існування одних без інших неможливе.

Склад програмного забезпечення називається програмною конфігурацією.

Склад програмного забезпечення утворює так звану піраміду (ієрархію). Програмне забезпечення будь-якого комп'ютера має таку піраміду, на найнижчому рівні якої знаходиться *базове програмне забезпечення*.

Базове програмне забезпечення (БПЗ) – забезпечує взаємодію базових апаратних засобів. БПЗ входить в склад базового апаратного обладнання і зберігається в ньому у мікросхемах («прошивається» в мікросхемах).

Системне ПЗ (програмне забезпечення):

- 1) *Драйвери* – програми, які забезпечують взаємодію програм з комп'ютерним обладнанням. Наприклад, якщо вмикається новий пристрій на системному рівні, потрібно встановити програму, яка буде забезпечувати взаємодію інших програм з цим пристроєм.
- 2) Програми, які забезпечують користувачу доступ до комп'ютера і можливість керувати його роботою (наприклад вводити дані) – *засоби інтерфейсу користувача*.

Службове ПЗ (програмне забезпечення) (утиліти) – відповідає за перевірку, налаштування та налагодження комп'ютерної системи. До цього класу ПЗ відносяться:

- диспетчери файлів;
- архіватори;
- перегляди та відтворення;
- контролю.

Прикладне ПЗ (програмне забезпечення) забезпечують виконання на комп'ютері конкретних завдань. До прикладного ПЗ відносяться:

- 1) Текстові редактори.
- 2) Графічні редактори.
- 3) Електронні таблиці.
- 4) Системи автоматизованого проектування (*CAD-системи*).
- 5) Редактори *HTML (Web –редактори)*.

Операційна Система (ОС) – комплекс системних та службових засобів. Основна функція ОС – посередництво; *ОС має забезпечувати діалог:*

- користувача і командно-апаратних засобів (інтерфейс користувача);
- програмного і апаратного забезпечення (апаратно-програмний інтерфейс);
- між різними видами програмного забезпечення (програмний інтерфейс).

Інтерфейс користувача – це управління комп'ютером, яке здійснює користувач.

Інтерфейс користувача розрізняють:

- *неграфічний* (задання команд комп'ютеру за допомогою клавіатури – або в виді стрічки символів на мові програмування (Basic, Fortran і ін.); характерний для операційних систем із назвою *MS-DOS*.

- *графічний* інтерфейс (управління комп'ютером здійснюється, як за допомогою клавіатури так і за допомогою графічного елемента екрану – курсору миші).

Графічний елемент екрану *курсор миші* – це *активний елемент управління*.

Графічні елементи управління програмами (*екранні кнопки, значки, перемикачі, розгорнуті списки, стрічки меню* и т.д.) – це *пасивні елементи управління*.

ОС Windows – це графічна система. Головна ідеологія ОС Windows – забезпечити графічний інтерфейс користувача таким чином, щоб максимально спростити користувачу управління комп'ютером.

Біл Гейтс: Комп'ютером повинна вміти керувати домогосподарка.

Робочий стіл Windows

Робочий стіл Windows – стартовий екран.

На *робочому столі* відображаються елементи *Windows* і елементи управління *Windows* (Рис. 2).

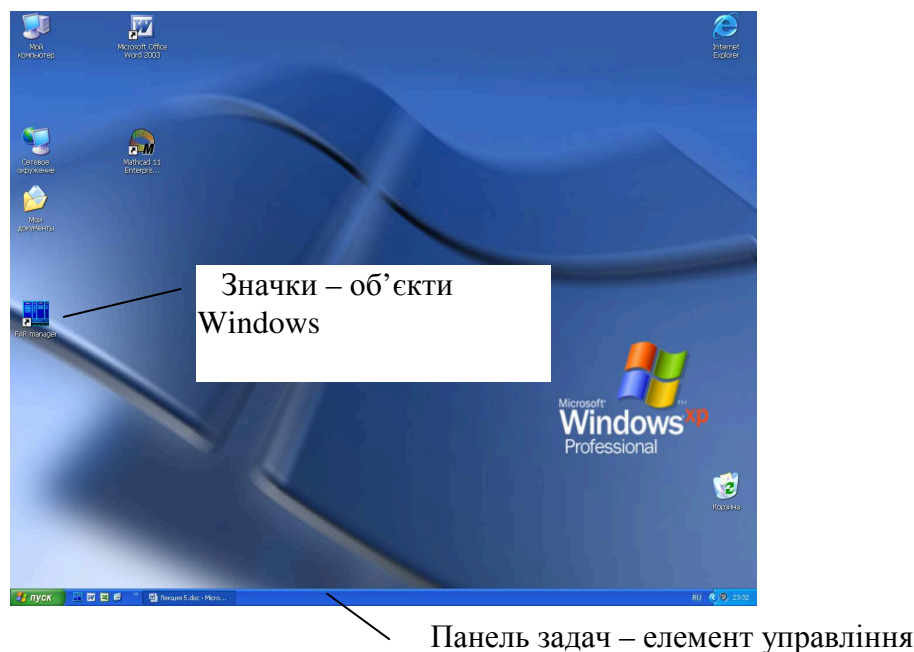


Рис.2 – Робочий стіл Windows

Управління комп'ютером в ОС Windows за допомогою курсора миші здійснюється за допомогою таких прийомів:

- 1) натискання (click);
- 2) подвійне клацання;
- 3) натискання правої клавіші – викликання контекстного меню;
- 4) перетягування
- 5) протягування;
- 6) зависання – поява підказки.

Ці прийоми управління поряд із значками, стрічки панелі задач Windows також дають змогу полегшити користувачу діалог з комп'ютером. Це стосується таких прийомів як використання клацання правої клавіші миші або зависання. При використанні цих прийомів на екрані з'являються розгорнуті меню команд, які є дозволеніми при роботі з об'єктами і з'явлення підказки.

Структура вікна текстового редактора Word

Розглянемо структуру вікна текстового редактора Word (Рис.3).

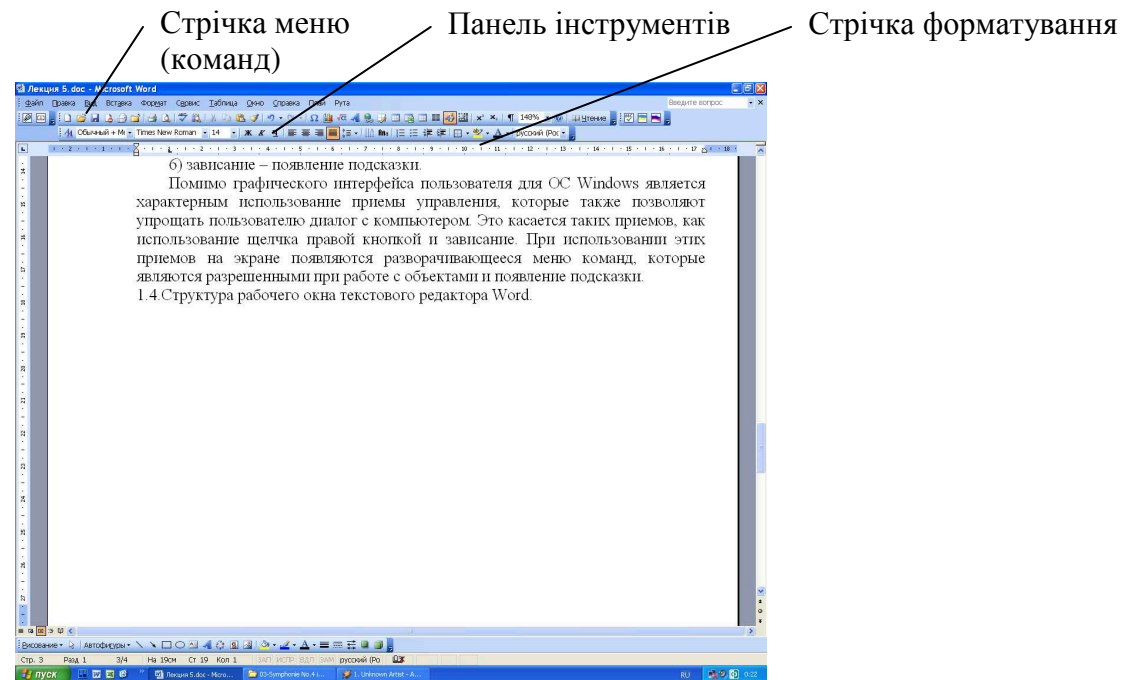


Рис. 3 – Структура вікна текстового редактора Word 2003

Стрічка меню (команд) – самий верхній ряд у ленті вікна текстового редактора Word 2003. Наприклад: Файл, Правка, Вид і т.д. У більш пізніших версіях (Word 2010 і далі) стрічка меню містить інші назви команд: Главная, Вставка, Разметка страницы і т.д.

Панель інструментів – нижче ряд у ленті вікна текстового редактора Word 2003. Наприклад: Створити документ (значок порожньої сторінки), Зберегти документ (значок дискети), Друк документу (значок принтеру). У більш пізніших версіях (Word 2010 і далі), панелі інструментів знаходять клацанням лівої кнопки миші на лівий верхній кут – на кнопку Office в Word.

Стрічка форматування – це самий нижній ряд у ленті вікна текстового редактора Word. Наприклад у Word 2007 і пізніших версіях це такі вкладки: Буфер обміну, Шрифт, Абзац, Стили, Редактирование.

Література: [осн. 1, 2].

Завдання на СРС: Вивчення матеріалу лекції. Завдання на практикум з основних прийомів роботи з графічним інтерфейсом ОС Windows (зависання, протягування, перетягування, виділення). Поняття діалогового вікна, контекстного меню; головні прийоми роботи з правою та лівою кнопками графічного маніпулятора. Огляд стандартних Додатків Windows. Службові Додатки Windows: архіватори, буфер обміну. Стандартні засоби мультимедіа. Текстовий редактор Microsoft Word. Призначення та прийоми роботи з Microsoft Word. Структура службового вікна Microsoft Word, команди та інструменти Microsoft Word. Налаштування Microsoft Word. Створення комплексних текстових документів за допомогою Microsoft Word. Редактор формул. **Література:** [осн. 1, 2].

Лекція 6

Тема: Побудова графіків та діаграм редактором Excel. Розв'язання рівнянь чисельними методами; метод підбору за допомогою операції відділення кореню.

Створення електронних таблиць та побудова графіків редактором Excel

Редактор Excel призначений для створення електронних таблиць та роботи з ними. У редакторі Excel також можна будувати графіки та діаграми, а також виконувати обчислення у автоматичному режимі.

Документ, що створюється в Excel, називається **робочою книгою** (Рис. 4).

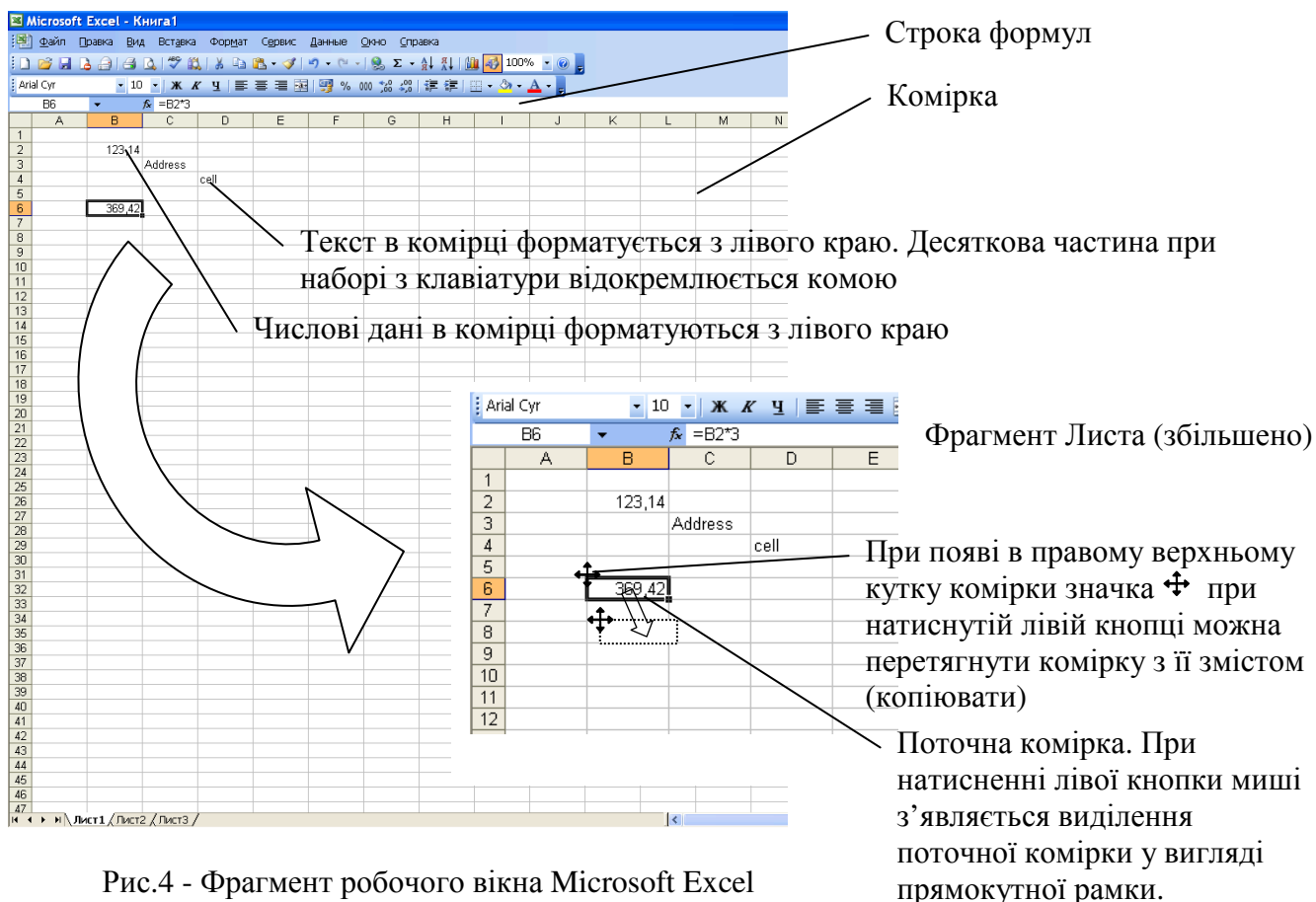


Рис.4 - Фрагмент робочого вікна Microsoft Excel

А. Адресація комірок. Адреси комірок в робочому листі Microsoft Excel містять дві координати: буквене значення стовпчика та номер строки. Показана на рисунку 4 комірка має адресу B6.

В. Діапазон комірок. При необхідності виділити діапазон даних, занесених у комірки можна за допомогою методу протягування вказівника миші при натиснутій лівій кнопці, як показано на рисунку 5.

	A	B	C	D	E	F	G
1	Time, hours	Time0, hot	Power, kW	Voltage, U	Current, I	CosF	
2	14	0	0	1500	188	9	0,97
3	14,5	0,5	1	1650	174	10	0,96
4	15	1	1	1650	174	10	0,96
5	15,5	1,5	2	1800	174	12	0,96
6	16	2	2	1850	168	12	0,96
7	16,5	2,5	3	1950	160	13	0,96
8	17	3	3	1950	160	13	0,96
9	17,5	3,5	4	2100	160	14	0,96
10	18	4	4	2100	160	14	0,96
11	18,5	4,5	5	2250	160	15	0,96
12	19	5	5	2250	160	15	0,96
13	19,5	5,5	6	2400	160	15	0,96
14	20	6	6	2400	160	15	0,96
15	20,5	6,5	7	2550	166	16	0,95
16	21	7	7	2550	166	16	0,95
17	21,5	7,5	8	2700	170	16	0,95
18	22	8	8	2700	170	16	0,95
19	22,5	8,5	9	2850	174	17	0,95
20	23	9	9	2850	174	17	0,94
21	23,5	9,5	10	3000	182	18	0,94

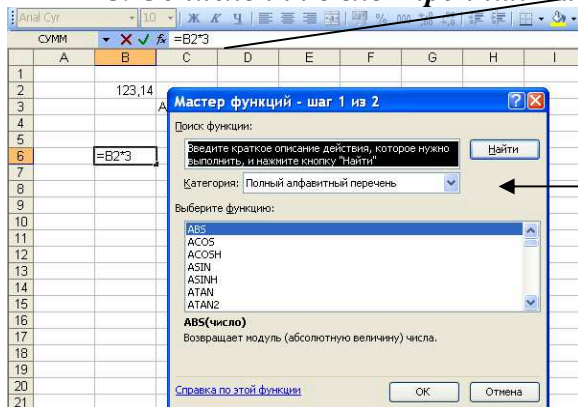
Виділений діапазон комірок. Позначається в документах Excel наступним чином: A2:E16

Рис.5 - Фрагмент робочого вікна Microsoft Excel виділення діапазону комірок

Працювати з документами Microsoft Excel потрібно при встановленому у вікні Панелі індикації англійської мови. Перевіряйте установку мови для того, щоб уникнути можливих помилок!!!



С. Обчислення в електронних таблицях.



Формули можуть містити числа, посилання на комірки (їх адреси) та функції Microsoft Excel (Рис.6).

Для виклику функцій та використання їх у формулах використовується *Мастер функций*, вид якого позначено на малюнку.

Головним правилом використання формул є потреба максимально використовувати можливості автоматичних обчислень у Microsoft Excel.

Рис.6 - Фрагмент робочого вікна Microsoft Excel обчислення в електронних таблицях

С1. Посилання на комірки. При автоматичних обчисленнях в Excel, формули, як правило, містять адреси комірок. Тобто формули містять посилання на адреси комірок, в яких містяться дані, необхідні для обчислень у формулі (Рис. 7).

	A	B	C	D	E
1					
2		123,14			
3			Address		
4			cell		
5					
6		=B2*3			
7					
8					

Рис.7 - Фрагмент робочого вікна Microsoft Excel посилання на комірки при обчисленнях

Посилання на комірки за замовченням розглядаються Excel як відносні (Рис. 8):

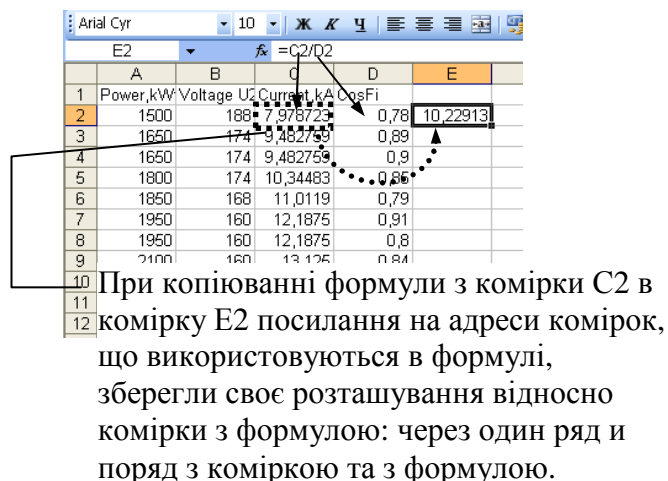
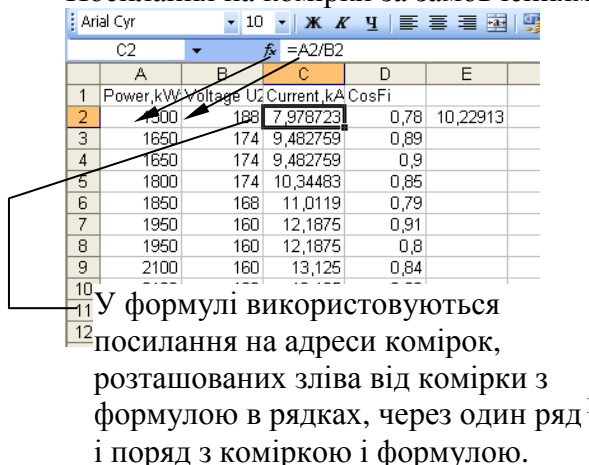


Рис.8 - Фрагмент робочого вікна Microsoft Excel з застосуванням відносних посилань на комірки при обчисленнях та при копіюванні

Адреса комірки у посиланнях у формулах остається незмінною, в якому б місці листа не знаходилася копія формули, для цього використовується абсолютна адресація комірок. Такі посилання називаються *абсолютними*.

Для створення абсолютних посилань в адресу комірки необхідно вписати символ \$. Символ \$ вводиться з клавіатури при одночасному натисненні клавіш (Shift "4").

Можливі наступні варіанти закріплення в адресах абсолютних посилань:

- \$A2 - абсолютне посилання на ряд A
- A\$2 - абсолютне посилання на строку 2
- \$A\$2 - абсолютне посилання на комірку A2.

D. Автоматизація вводу даних

- Автозаповнення числами:

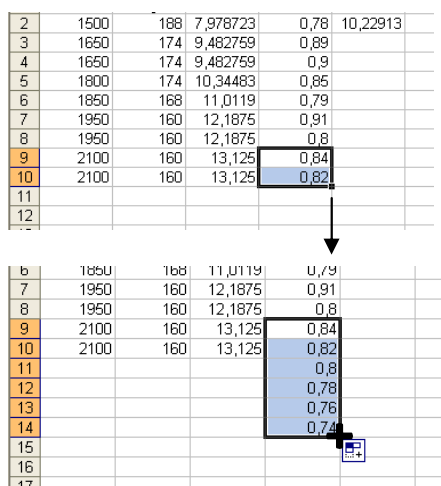
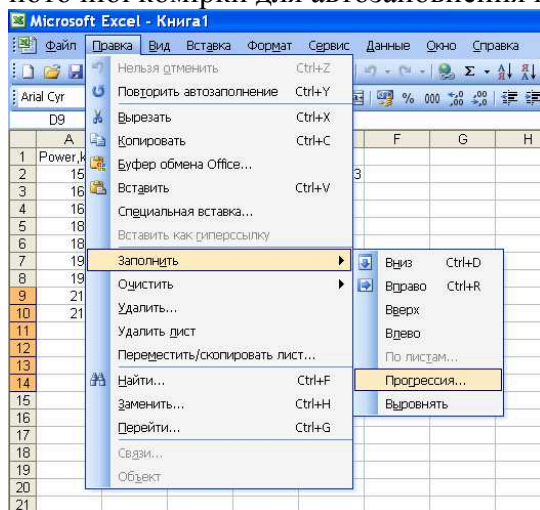


Рис.9 - Фрагмент робочого вікна Microsoft Excel з автозаповненням числами

Напрямок протягування при автозаповненні можна втілювати і в вертикальному і в горизонтальному положенні робочого!!!

Перевірте на практиці, що буде отримано при горизонтальному протягуванні поточної комірки для автозаповнення горизонтального ряду.



При необхідності змінити параметри автозаповнення потрібно використати меню (Рис.10):

Правка ► Заполнить ► Прогрессия...

Після виклику контекстного меню Прогрессия – вибрати тип прогресії та величину кроку

Рис.10 - Фрагмент робочого вікна Microsoft Excel для зміни параметрів автозаповнення

- Автозаповнення формулами:техніка заповнення така ж сама, як і числами – методом протягування. Однак для того, щоб посилання в формулах були вказані вірно потрібно використати вказівками про абсолютні та відносні посилання (розділ C1).

Література: [осн. 1, 2].

Завдання на СРС: Вивчення матеріалу лекції. Завдання на комп'ютерний практикум з основних прийомів роботи з редактором Microsoft Excel. Методичні вказівки до створення протоколу комп'ютерного практикуму. **Література:** [осн. 1, 2].

Лекція 7

Тема: Розв'язання математичних задач на базі лінійних та нелінійних рівнянь за допомогою Excel.

Методи наближеного розв'язку рівнянь в редакторі Excel

Редактор Excel дозволяє достатньо легко вирішати алгебраїчні рівняння. Розглянемо можливості розв'язку рівнянь з допомогою програми Microsoft Excel. Перш, ніж приступати до пояснення методів розв'язків рівнянь в Microsoft Excel згадаємо, що згідно з математикою називається розв'язком рівняння.

Будемо називати рівнянням вираз виду $f(x) = 0$, де $f(x)$ окремо являє собою функцію незалежного аргументу, тобто x . Тоді корінь рівняння – це таке значення аргументу, яке перетворює вираз у нуль. Розв'язати рівняння – означає знайти таке значення кореня. В математиці прийнято називати точним розв'язком рівняння таке, яке може бути представлене в виді формул. Прикладом такої функції є наприклад формула для розв'язку квадратного рівняння $ax^2 + bx + c = 0$:

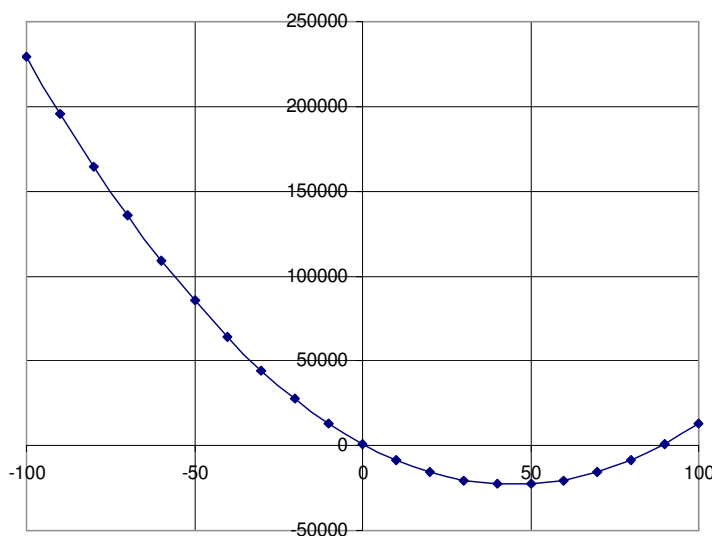
$$x_1, x_2 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Тільки для дуже обмеженого діапазону функцій є методики для знаходження точного кореня рівняння. Але числове значення кореня не завжди може бути зробленим правильно.

Повернемось до визначення рівняння і його коренів. Отож, якщо $f(x) = 0$ - рівняння, а $f(x)$ - функція, то із визначення видно, що корінь рівняння – це таке число, при якому функція перетворюється у нуль – тобто перетинає вісь X . Нижче на рис.7.1. зображений графік функції $f(x) = 12x^2 - 1080x + 1375$. Із графіка видно, що в діапазоні значень $(-100 < x < 100)$ він перетинає вісь X двічі, тобто рівняння $12x^2 - 1080x + 1375 = 0$ має як мінімум два корені.

Графічний метод пошуку кореня рівняння

Із сказаного ясно, що коли побудувати графік функції можна легко побачити, чи є корені і скільки їх. Для початкової оцінки розв'язку рівняння це вжене поганий результат. Дуже часто так так і поступають навіть при необхідності отримання більш точних розв'язків: будують графік функції (або по крайній мірі ставлять знаки функції на інтервалі визначення).



На рис.11 видно, що рівняння $12x^2 - 1080x + 1375 = 0$ має два корені, тому що графік перетинає вісь X в двох точках.

По рис.11 ми можемо визначити корені рівняння тільки приблизно.

Графічний метод пошуку коренів рівняння є оцінкою в першому наближенні розв'язку рівняння.

За допомогою графіка ми можемо наглядно переконатись скільки коренів у рівняння функції і чи є вони взагалі.

Рис.11 - Графік функції $y(x) = 12x^2 - 1080x + 1375$

Для більш точного визначення коренів можна збільшити масштаб осі X і визначити у збільшеному масштабі осі X , де графік функції перетинає її, як показано на рис.12.

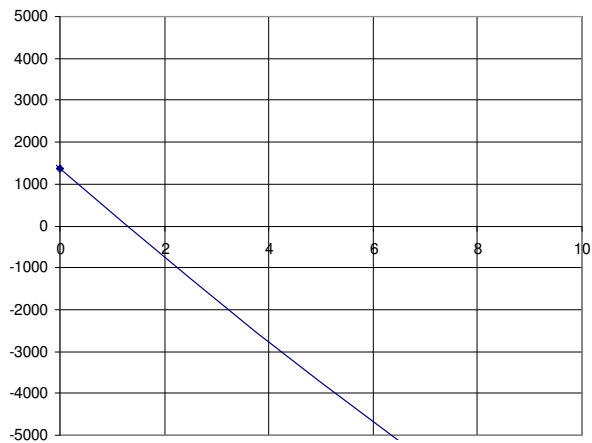
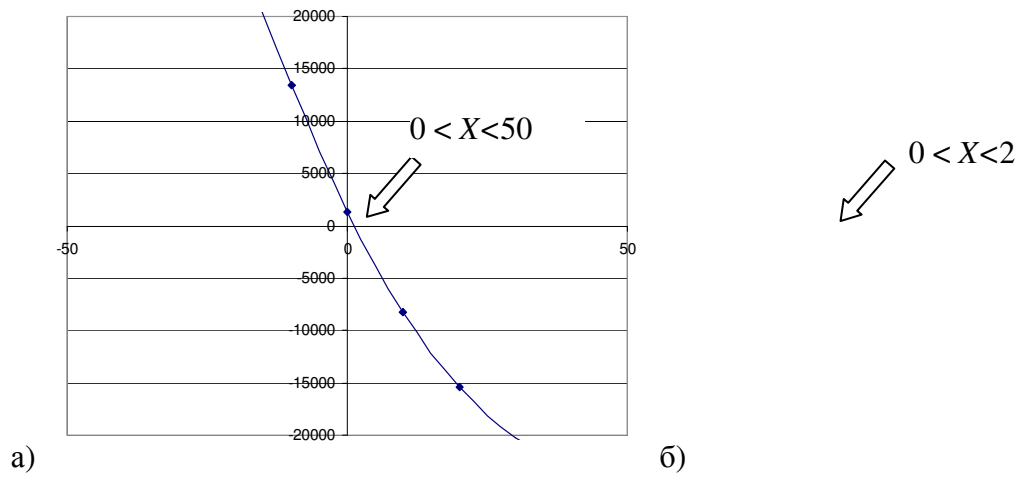
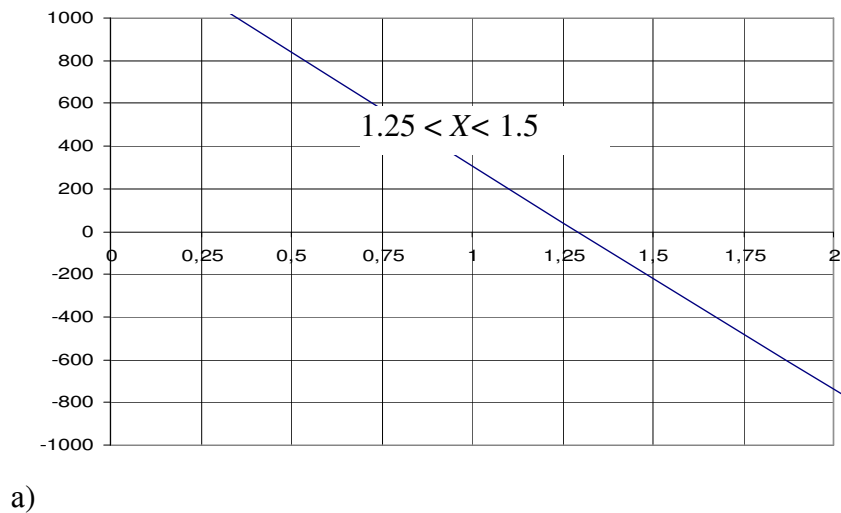
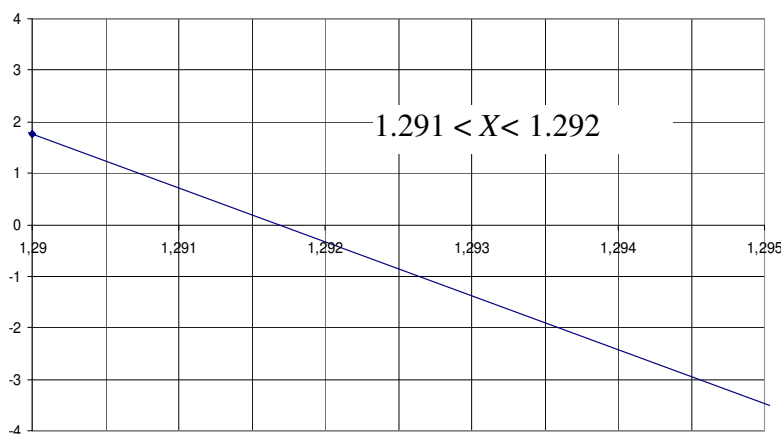


Рис.12 - Наближення кореня рівняння при зміні масштабу графіка

На рис.12(б) вже ясно, що один із коренів знаходиться на інтервалі $[0,2]$. Подальша зміна масштабу дозволить ще більше уточнити значення шуканого кореня.





б)

Рис.13 - Уточнення приблизного значення кореня

На рис.13(б) видно, що наближене значення кореня знаходиться на інтервалі $[1,2915, 1,2920]$. Таке вирішення вже є досить непоганим наближенням. Значення функції при вказаних значеннях кореня $[1,2915, 1,2920]$ відповідно дорівнюють $y(1,2915) = 0.195667$, $y(1,2920) = -0.32883$.

Аналогічно можна отримати і значення другого кореня.

Наведений приклад демонструє графічний метод пошуку кореня рівняння.

Графічний метод пошуку кореня дозволяє наглядно отримати уявлення про наявність коренів рівняння та наближено визначити їх.

Важливо підмітити, що при дуже малому масштабі графіку можна втратити графічне зображення. Наприклад, на стадії ділення інтервалу значень x , як на рис.13(б) можна не отримати графічного зображення. Така ситуація може виникнути, якщо крок по x вибрано достатньо великим, наприклад, рівним 1. Для того, щоб отримати більш точне графічне зображення, потрібно вибрати менший крок по x .

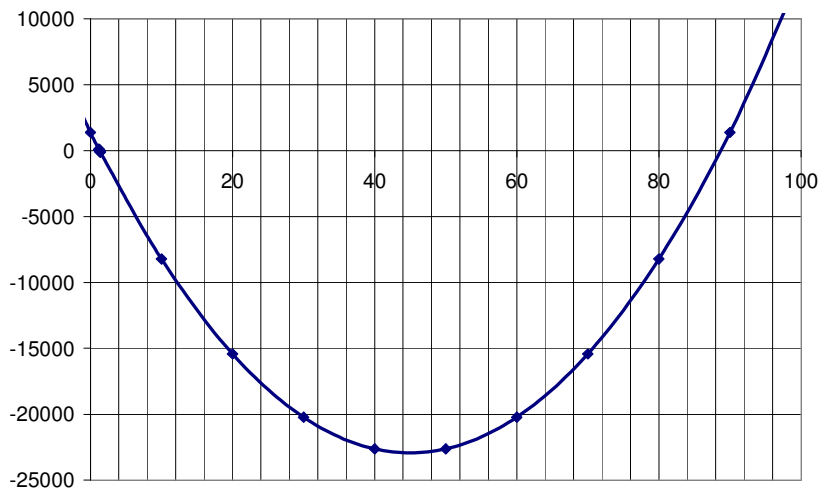
Однак, графічний метод, як це видно не дає точного розв'язку рівняння. Що це означає?

Для більш точного розв'язку рівняння, як відомо, потрібно спочатку задати ту точність яку нам потрібно.

*Розв'язування рівняння редакторі Excel з допомогою режиму «Формула»
(комбінований метод)*

Розв'язування подібне тому, яке ми розглядали при графічному методі.

Розв'язування розпочинаємо з побудови графіка для того, щоб впевнитись в тому, що дане рівняння має корені. Одночасно при побудові графіка перевіряємо, скільки коренів може мати рівняння. На цьому етапі визначаються інтервали знаходження коренів з точністю до одиниць.



Із рис.14 беремо у першому наближенні корені даної функції у діапазонах:

$$0 < x_1 < 5$$

$$82 < x_2 < 90$$

Рис.14 - Графік функції $y(x) = 12x^2 - 1080x + 1375$

Виділяємо на Листі Excel комірки для вирішення рівняння. Нехай це будуть комірки A1 і B1.

В комірку A1 записуємо початкове значення інтервалу, в якому заходиться початковий корінь, тобто 0. Вибираємо крок зміни по x . Нехай крок по x буде 0,5.

В комірку A2 записуємо наступне значення $x = 0,1$. Методом протягування заповнюємо комірки в стовпці A до значення $x = 5$.

В комірку B1 записуємо формулу для функції $=12x^2 - 1080x + 1375$

Методом протягування копіюємо формулу із комірки B1 в інші комірки стовпця B. Отримаємо наступний результат:

	A	B	C	D	E
1	0	1375			
2	0,5	838			
3	1	307			
4	1,5	-218			
5	2	-737			
6	2,5	-1250			
7	3	-1757			
8	3,5	-2258			
9	4	-2753			
10	4,5	-3242			
11	5	-3725			
12					

Із фрагмента листа Excel видно, що перехід через 0 функції $f(x) = 12x^2 - 1080x + 1375$ відбувається на інтервалі (1; 1,5). Тобто корінь $1 < x < 1,5$ (Рис. 15)

Далі точно так само як ми ділили інтервал знаходження кореня (0;5) ділимо інтервал (1;1,5).

Вибираємо крок по $x=0,01$ і виконуємо автоматичне визначення коренів рівняння. В другому наближенні отримаємо результат, показаний на рис.15

Рис.15 - Наближений розв'язок рівняння $y(x) = 12x^2 - 1080x + 1375$ в діапазоні x від 0 до 5

Фрагмент автоматичних обчислень наведений на рис.16 звідки видно, що друге наближення дало інтервал для кореня $1,29 < x < 1,30$.

	A	B	C	D
103	1,2	96,28		
104	1,21	85,7692		
105	1,22	75,2608		
106	1,23	64,7548		
107	1,24	54,2512		
108	1,25	43,75		
109	1,26	33,2512		
110	1,27	22,7548		
111	1,28	12,2608		
112	1,29	1,7692		
113	1,3	-8,72		
114	1,31	-19,2068		

Отримане наближення вже є достатньо хорошим для визначення кореня рівняння. Однак із фрагмента обчислень видно, що функція на границях інтервалу змінюється дуже суттєво.

Рис.16 - Наближений розв'язок рівняння $y(x) = 12x^2 - 1080x + 1375$ в діапазоні x від 1,2 до 1,30

Подальші уточнення будемо вести так, щоб значення функції якомого більше відрізнялось від нуля. Нехай розв'язком рівняння буде таке число x_0 , щоб $|y(x_0) - 0| < \varepsilon$, де $\varepsilon \ll 1$. Нехай $\varepsilon = 0,001$.

	A	B	C
103	1,291	1,7692	
104	1,291	0,720172	
105	1,292	-0,32883	
106	1,293	-1,37781	
107			
108			
109			
110			

$$1,291 < x_0 < 1,292$$

	A	B	C
103	1,291	0,720172	
104	1,2911	0,615271	
105	1,2912	0,510369	
106	1,2913	0,405468	
107	1,2914	0,300568	
108	1,2915	0,195667	
109	1,2916	0,090767	
110	1,2917	-0,01413	
111	1,2918	-0,11903	
112	1,2919	-0,22393	

$$1,2916 < x_0 < 1,2917$$

	A	B	C
103	1,2916	0,090767	
104	1,29161	0,080277	
105	1,29162	0,069787	
106	1,29163	0,059297	
107	1,29164	0,048807	
108	1,29165	0,038317	
109	1,29166	0,027827	
110	1,29167	0,017337	
111	1,29168	0,006847	
112	1,29169	-0,00364	

$$1,29168 < x_0 < 1,29169$$

	A	B	C
103	1,29168	0,006847	
104	1,291681	0,005798	
105	1,291682	0,004749	
106	1,291683	0,0037	
107	1,291684	0,002651	
108	1,291685	0,001602	
109	1,291686	0,000553	
110	1,291687	-0,0005	
111	1,291688	-0,00155	
112	1,291689	-0,00259	

$$1,291686 < x_0 < 1,291687$$

Рішення знайдено: $|0,000553 - 0| < 0,001$

Рис.17 - Наближений розв'язок рівняння методом послідовних наближень за формулою

Розв'язок рівняння в Excel в автоматичному режимі

Програма Excel дозволяє знаходити корені нелінійних рівнянь в автоматичному режимі.

Для цього в меню команд є команда «Подбор параметра»:

Строка команд ► Сервис ► Подбор параметра

При виклику цієї команди, Excel перебирає по спеціальному алгоритмі числа і підставляє їх послідовно в формулу, яка написана в певній комірці таким чином, щоб в цій комірці було отримано задане число.

Розв'язування рівняння виконуємо наступним чином:

А) Розв'язування розпочинаємо з побудови графіка для того, для того щоб впевнитись у тому, що дане рівняння має корені. Одночасно при побудові графіка перевіряємо, скільки коренів може мати рівняння. Кільки коренів може мати рівняння на цьому етапі визначаємо інтервали з точністю до одиниць.

Б) В комірці B1 записуємо рівняння $=12x^2 - 1080x + 1375$

В) В комірці A1 записуємо перше наближене значення (наприклад, в нашому випадку 0).

Г) Викликаємо діалогове меню: Строка команд ► Сервис ► Подбор параметра. Отримаємо вид вікна як зображено на рис.18.

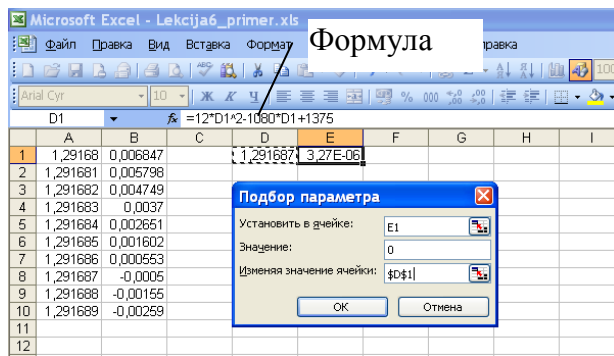
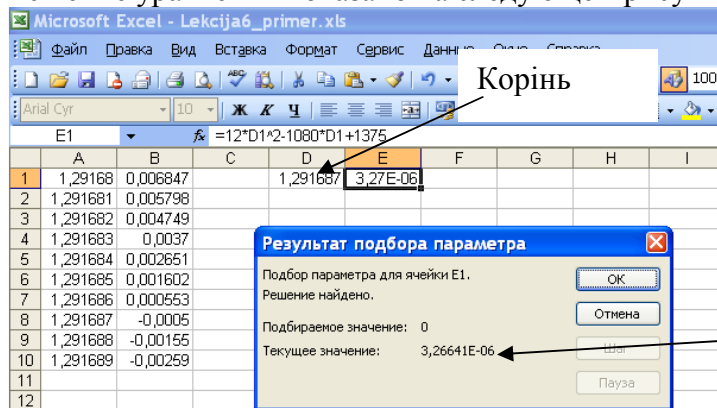


Рис.18 - Контекстне меню «Підбір параметру»

В комірці E1 записано рівняння $12x^2 - 1080x + 1375 = 0$ в вигляді формули. Якщо в комірці буде отримано значення 0, то в комірці D1 буде число, яке є коренем рівняння.

Тому автоматичний пошук кореня виконуємо так, як показано на рис. 18-19

Решение уравнения показано на следующем рисунке



Як видно із співставленням з отриманим раніше розв'язком, отриманий методом підстановки(Рис.19) рішення дало близький результат

Значення функції близьке до 0.

Рис.19 - Розв'язок нелінійного рівняння $12x^2 - 1080x + 1375 = 0$ в автоматичному режимі

Математичні основи числового обчислення рівнянь

Описаний метод розв'язку рівняння носить назву метода спроб. Він відноситься до так званих ітераційних методів. Суть методу ітерацій заключається в повторному застосуванні якої-небудь математичної операції. Математичні основи метода заключаються у багаторазовому повторенні ітерацій, що називається виділенням кореня.

Нехай задано рівняння

$$f(x) = 0,$$

де функція $f(x)$ визначена і нерозривна на відрізку (a,b) . Умова нерозривності необхідна для того, щоб бути впевненим, що функція не втратить монотонного характеру змін.

Для того, щоб вияснити, чи є на будь-якому (попередньо невідомого) відрізку корінь рівняння, необхідно щоб виконувались дві умови:

а) Метод прямих ітераційних обчислень кореня.

Нехай на кінцях відрізка функція має різні знаки; для відрізка $[a,b]$ має бути $f(a) \cdot f(b) < 0$.

Геометрично це означає, що графік функції $f(x)$ в точках a і b знаходиться по різні сторони від осі x . Зрозуміло, що в середині відрізка графік обов'язково буде перетинати вісь x .

Математики полюбляють застосовувати терміни, які погано пояснюються. Так наприклад постановці задачі говориться: нехай для рівняння відомий інтервал відділення кореня. Що це означає? Це означає, що шляхом методу проб, багаторазових обчислень, створення таблиць значень функції, побудови графіків ми побачили, що дійсно на інтервалі $[a,b]$ функція міняє знак (Рис.20). Всю ретельну і об'ємну роботу по візуалізації цього інтервалу математики не згадують – вони відразу говорять про готовий результат – нехай відомий інтервал відділення кореня.

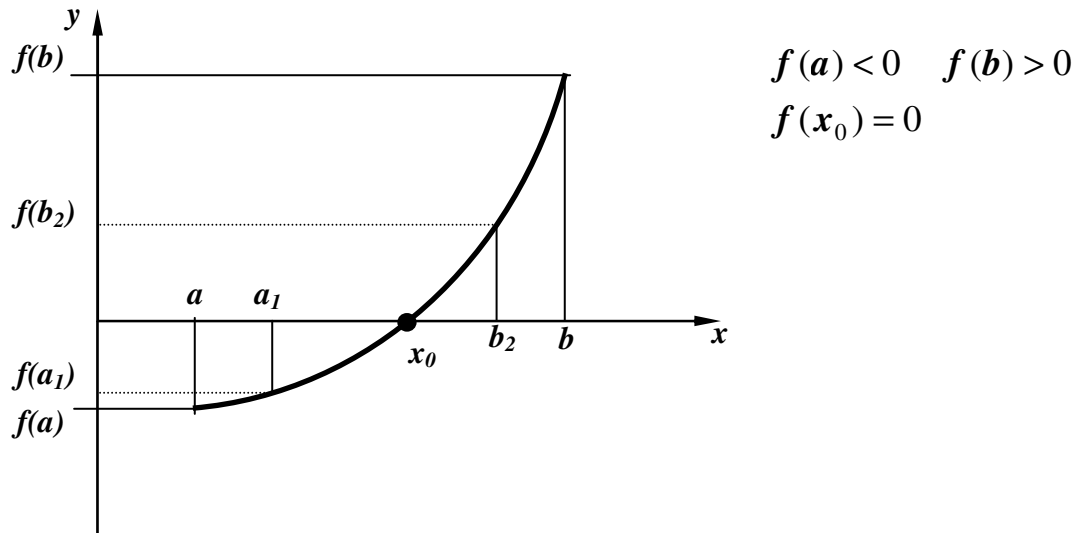


Рис.20 - Розв'язок нелінійного рівняння на інтервалі $[a,b]$

Умова знаходження кореня при послідовному звуженні інтервалу відділення кореня $[a,b], [a_1,b_1]$ і т.д.) заключається в тому, щоб знайти такий інтервал $[a_n,b_n]$, для якого виконувались би вимоги: $|f(a_n) - f(b_n)| < \epsilon$, де ϵ певне, наперед задане мале число ($\epsilon \ll 1$). Тобто різниця $|f(a_n) - f(b_n)|$ мало відрізняється від нуля. Це означає, що істинне значення функції, рівне нулю $f(x_0) = 0$, яке може бути отримане при шуканому (x_0) знаходиться в знайденому інтервалі $|f(a_n) - f(b_n)|$, а власне корінь рівняння – в інтервалі $[a_n,b_n]$. Шукане значення вважається рівним

$$x_0 = \frac{a_n + b_n}{2}.$$

Описаний метод може бути використано при автоматичних обчисленнях в редакторі Excel. Практичне використання даного методу дозволяє достатньо швидко знаходити корені рівнянь з одною невідомою.

Недоліком методу є відсутність надійного алгоритму обчислення, що робить метод не повністю придатним до автоматичного визначення.

Література: [осн. 1, 2,].

Завдання на СРС: Вивчення матеріалу лекції. Завдання на комп'ютерний практикум з основних прийомів роботи з редактором Microsoft Excel. Видача варіантів розв'язання рівнянь. Література: [осн. 1, 2].

Лекція 8

Тема: Принципи математичних обчислень у редакторі MathCad

Головною позитивною відмінністю програми MathCad є те, що для реалізації будь-яких математичних операцій, обчислень не потрібно освоювати спеціальні мови програмування. Тобто в MathCad використовується одна універсальна математична символіка, яка відома з курсів арифметики і алгебри. Запис формул, математичних виразів така ж, як це прийнято при записі на аркуші паперу.

Програма MathCad містить вбудовану внутрішню мову програмування, яка дозволяє розрізняти символи математичних виразів і виконувати з ними необхідні дії.

Вхідна мова MathCAD відноситься до інтерпретаторського типу. Це означає, що коли вона пізнає який-небудь об'єкт системи, то тут же виконує зазначені в блоці операції. Мовою реалізації системи є мова C++.

У міру того як користувач створює (засобами текстового, формульного і графічного редакторів) у вікні редагування об'єкти (тексти, формули, таблиці та графіки), система сама складає програму на деякому проміжному мовою зв'язку, яка зберігається в оперативному запису вальному пристрої (ОЗП) до тих пір, поки не буде збережена на диску у вигляді файлу з розширенням `xmcd`.

Підготовка обчислювальних блоків полегшується завдяки висновку шаблону при завданні того чи іншого оператора. Для цього в MathCAD служать складальні панелі з шаблонами різних математичних символів. Припустимо, потрібно обчислити визначений інтеграл. Для цього спочатку треба вивести панель операторів математичного аналізу; її піктограма в рядку інструментів має знаки інтеграла і похідної. Потім слід встановити візир в те місце екрану, куди виводиться шаблон, і на панелі зробити активною піктограму з зображенням знака визначеного інтеграла. У складі складних шаблонів часто зустрічаються шаблони для введення окремих даних. Вони мають вигляд невеликих чорних квадратиків. У шаблоні інтегралу їх чотири: для введення верхньої і нижньої меж інтегрування, для задання підінтегральної функції і для вказівки імені змінної, по якій йде інтегрування.

Текстовий редактор дозволяє задавати текстові коментарі. У простому випадку для відкриття текстового редактора достатньо ввести символ " (одиначна лапка). В з'явився прямокутник можна почати вводити текст. Mathcad визначає призначення поточного блоку автоматично при першому натисканні клавіші ПРОБІЛ. Якщо введений текст не може бути інтерпретований як формула, блок перетворюється в текстовий і наступні дані розглядають ся як текст. Створити текстовий блок без використання автоматичних засобів дозволяє команда `Insert • Text Region` (Вставка • Текстовий блок).

Знаком присвоювання в MathCAD є: `=` (вводиться як `:`). Припустимо застосовувати як знак присвоювання і `=`. Система автоматично замінює його на знак: `=` при першій операції присвоювання. Наприклад, якщо ви введете в перший раз `x = 2`, то система представить це у вигляді `x: = 2`. Однак при другому привласненні треба писати `x: = 3`

(якщо задати $x =$, то система відразу видасть $x = 2$, тобто знак $=$ буде вже означати висновок).

На рис.21 (а, б) показано робоче вікно MathCAD.

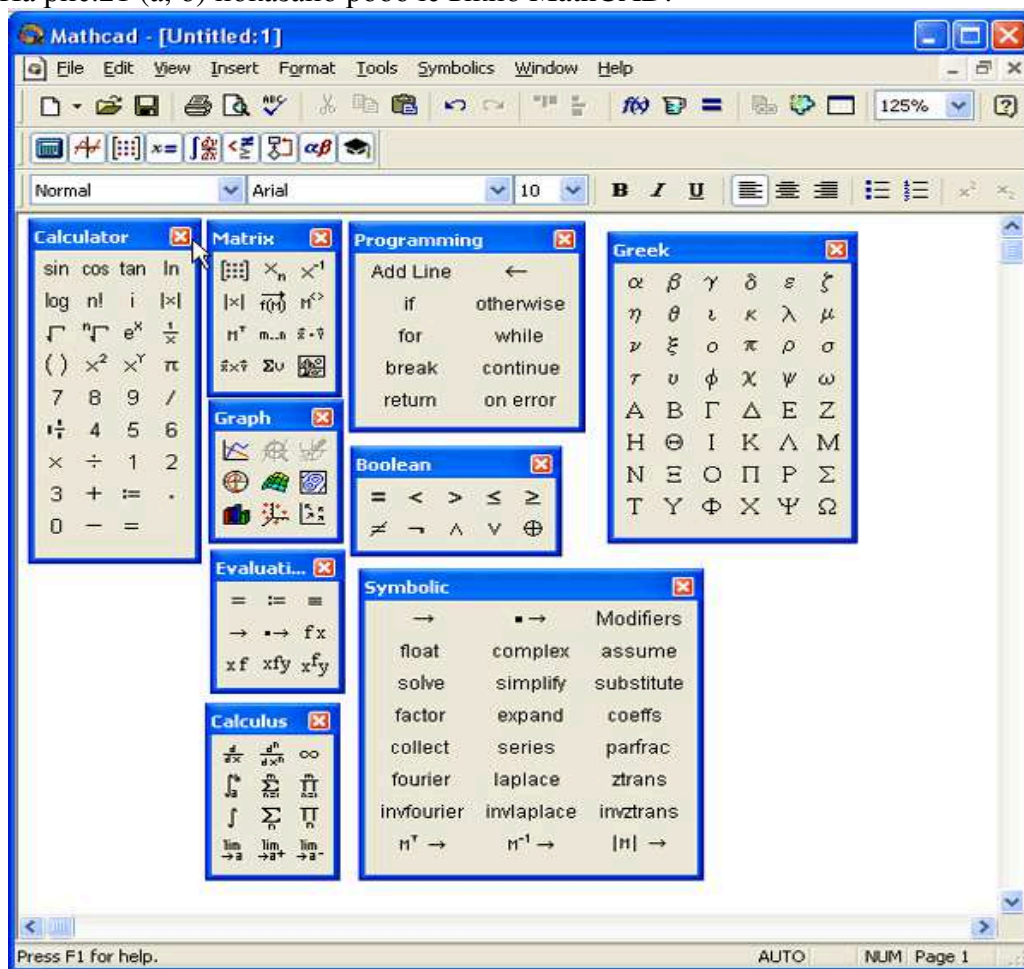


Рис. 21, а) Панель інструментів MathCAD










-  – меню Calculator (Калькулятор)
-  – меню Graph (График)
-  – меню Matrix (Матрица)
-  – меню Evaluation (Вычисление)
-  – меню Calculus (Исчисление)
-  – меню Boolean (Логический)
-  – меню Programming (Программирование)
-  – меню Greek (Греческий)
-  – меню Symbolic (Символьная)

Рис. 21, б) Піктограми панелі інструментів MathCAD

Щоб позначити об'єкт пунктирною лінією, потрібно встановити курсор близько вибраного об'єкту і, натиснувши праву клавішу миші, переміщувати її по столу. При цьому з'являється пунктирний прямокутник, який переміщенням миші можна розширювати в різних напрямках. Як тільки цей прямокутник захоплює один або кілька об'єктів, вони виявляються також виділеними - обведеними пунктирною лінією. Виділення пунктирною лінією використовується для переміщення об'єктів по вікну. Для цього потрібно всередину зазначеного об'єкта помістити курсор миші, натиснути праву клавішу і, утримуючи її натиснутою, переміщати мишу. Об'єкт (об'єкти) можна залишити, відпустивши клавішу миші на новому місці.

MathCAD має безліч вбудованих функцій, які володіють особливою властивістю: у відповідь на звернення до них по імені з вказівкою аргументу (або списку аргументів) вони повертають деяке значення - символічне, числове, вектор або матрицю. У систему вбудований ряд функцій, наприклад функція обчислення синуса $\sin(x)$ аргументу x , логарифма $\ln(x)$ і т. д. Поряд з вбудованими функціями можуть задаватися і функції користувача, відсутні в MathCAD. Звичайні змінні відрізняються від системних тим, що вони повинні бути попередньо визначені користувачем, тобто їм необхідно хоча б один раз присвоїти значення. В якості оператора присвоєння використовується знак: $=$, тоді як знак $=$ відведений для виводу значення чи константи змінної. Спроба використати невизначену змінну веде до висновку повідомлення про помилку - мінлива забарвлюється в яскраво-червоний колір.

Змінні можуть бути і розмірними, тобто характеризуватися не лише своїм значенням, але і вказівкою фізичної величини, значення якої вони зберігають. Для присвоювання значень таким змінним використовується знак $=$. Використовувана система одиниць відображається в діалоговому вікні Insert Unit (Вставка одиниць вимірювання).

Незважаючи на досить широкий набір вбудованих функцій, завжди виникає необхідність розширити систему новими функціями, котрі представляють інтерес для користувача. Функції користувача вводяться із застосуванням наступного виразу:

Ім'я_функції (Список_параметров): = Вираз

Ім'я функції задається як будь-який ідентифікатор, наприклад ім'я змінної. У дужках вказується список параметрів функції, це перелік використовуваних у вираженні змінних, поділених комами. Вираз - будь-який вираз, що містить доступні системі оператори і функції з операндами і аргументами, зазначеними в списку параметрів.

$\text{fun}(x) := 10 \cdot (1 - \exp(x)).$

$x := 5$

$v := 1 + \text{fun}(x)$

Функція умовних виразів if

Якщо для функції $y(x) = \sin(2 \cdot \pi \cdot x/3)$ умова перевіряється оператором *if* для діапазону значень змінної x від $-\pi$ до π і вона виконується, то буде обчислюватися вираз для цієї функції $y(x) = \sin(2 \cdot \pi \cdot x/3)$, а в іншому випадку оператором іншого вибору *otherwise* функції $y(x) = 0$, як видно на побудованому в MathCAD графіку цієї функції (Рис.22).

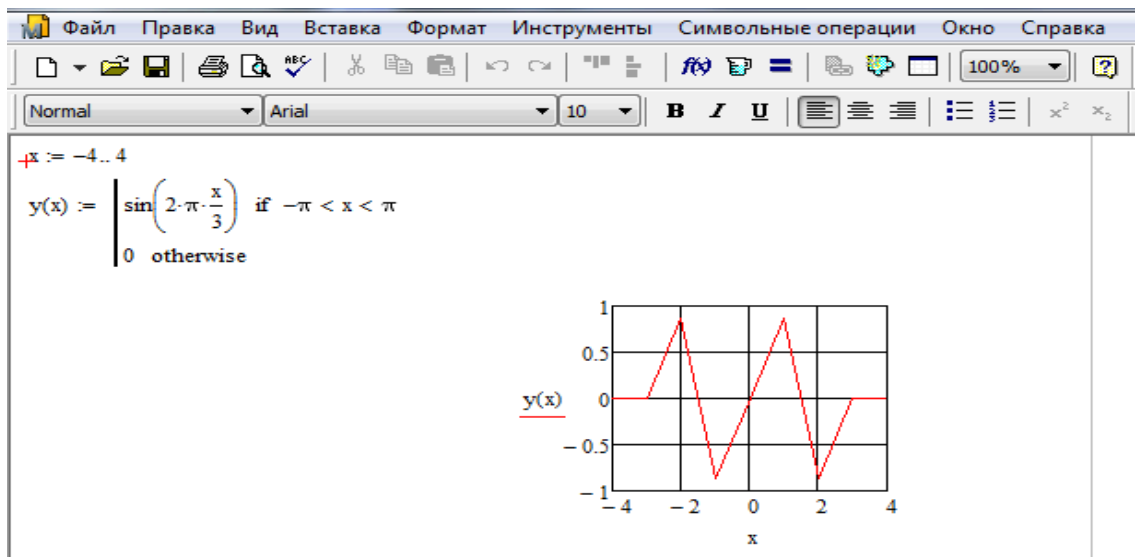


Рис.22 – Приклад застосування умовного оператора if для функції sin в Mathcad

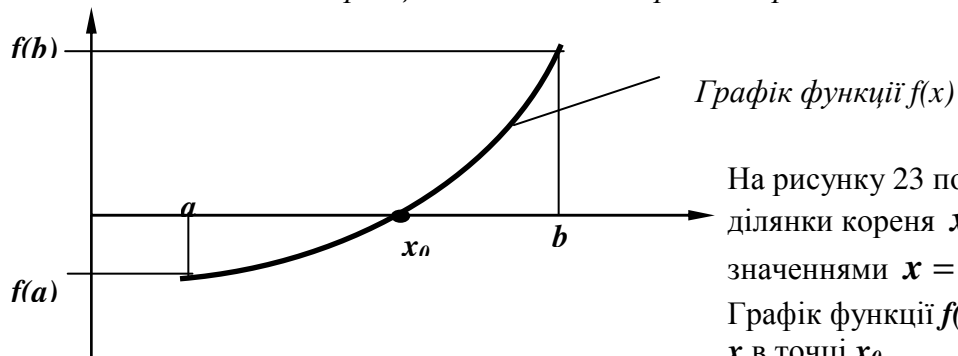
Література: [осн. 1, 4, дод. 2].

Завдання на СРС: Вивчення матеріалу лекції. Загальні відомості про класи обчислень в Mathcad. **Література:** [осн. 1, 4, дод. 2].

Лекція 9

Тема: Розв'язання лінійних та нелінійних рівнянь у MathCad наближеними методами (метод хорд і метод дотичних)

Основні математичні принципи наближеного рішення рівнянь методом хорд



На рисунку 23 показаний участок ділянки кореня x_0 . Ця ділянка між значеннями $x = a$ та $x = b$

Графік функції $f(x)$ перетинає вісь x в точці x_0 .

Рис. 23 - Графік функції $f(x)$ в

Основна думка: Нехай задана деяка функція $f(x)$, яка перетинає вісь x у точці x_0 (Рис.24) у вигляді математичного виразу. Цей вираз може бути як завгодно складним.

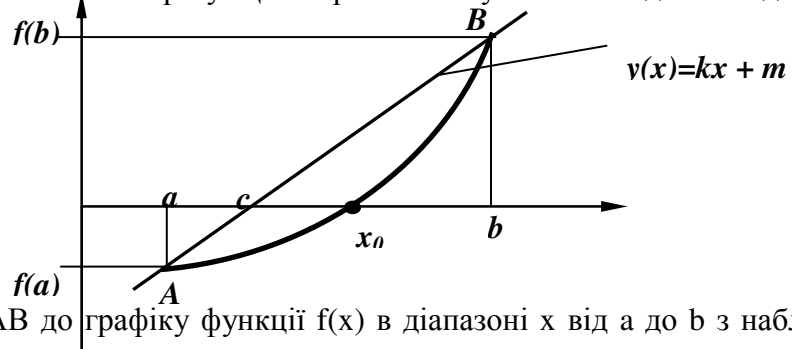


Рис.24 – Хорда АВ до графіку функції $f(x)$ в діапазоні x від a до b з наближеним коренем у точці c

Отже для його вирішення може не виявитися формули точного рішення (як наприклад таке точне рішення існує для квадратного рівняння тобто 2-го ступеню). Також ми знаємо точне рішення для лінійного рівняння $y(x) = k \cdot x + m$. Однак при першому погляді на графік вихідної функції $f(x)$ видно що він відрізняється від лінійної залежності для якої знайти точне рішення найбільш просто. Пропонується замінити ділянку графіка функції $f(x)$ лінійним відрізком між значеннями $x=a$ и $x=b$, як показано на рисунку 24.

Проведемо через крайні точки A і B графіка функції інтервала відділення кореня пряму (з побудови зрозуміло, що це – хорда для кривої Ax_0B). Хорда перетинає вісь x в деякій точці c . Зрозуміла що ця точка відстоїть до x_0 . Але все-таки вона знаходиться ближче до x_0 , ніж кінці інтервалу ab .

Тому будемо вважати точку c першим наближенням кореня. Чому нам зручно так вважати? Тому, що точка c лежить на лінії, заданої рівнянням $y(x)=ax + b$, корінь якого ми легко зможемо знайти, тобто отримати точне рішення.

Оскільки точки A і B належать обом лініям (тобто $f(x)$ і $y(x)$) можна зв'язати параметри обох залежних: висловити точне значення кореня рівняння $y(x)$ через параметри $f(x)$.

Запишемо систему двох рівнянь з двома невідомими (k і m – невідомі числа):

$$\text{при } x = a \quad y(a) = f(a) = A$$

$$\text{при } x = b \quad y(b) = f(b) = B$$

І, отже,

$$k \cdot a + m = f(a)$$

$$k \cdot b + m = f(b)$$

Звідки віднімаючи з (1) співвідношення (2) отримаємо:

$$k(a - b) = f(a) - f(b) \quad \text{І, відповідно, } k = \frac{f(a) - f(b)}{(a - b)}. \text{ А значення } m \text{ отримаємо}$$

після підстановки:

$$m = f(a) - k \cdot a = f(a) - \frac{f(a) - f(b)}{(a - b)} \cdot a$$

Для точки $x = c$ і, відповідно, $y(c) = 0$ отримаємо з рівняння лінійної функції:

$$y(c) = 0 = k \cdot c + m = \frac{f(a) - f(b)}{(a - b)} \cdot c + f(a) - \frac{f(a) - f(b)}{(a - b)} \cdot a$$

чи

$$\frac{f(a) - f(b)}{(a - b)} \cdot c + f(a) - \frac{f(a) - f(b)}{(a - b)} \cdot a = 0$$

Звідки отримаємо аналітичне значення кореня лінійної функції c .

$$\begin{aligned} \frac{f(a)-f(b)}{(a-b)} \cdot c + f(a) - \frac{f(a)-f(b)}{(a-b)} \cdot a &= 0 \\ \frac{f(a)-f(b)}{(a-b)} \cdot c &= \frac{f(a)-f(b)}{(a-b)} \cdot a - f(a) \\ c &= \frac{f(a)-f(b)}{(a-b)} \cdot \frac{(a-b)}{f(a)-f(b)} \cdot a - \frac{f(a)(a-b)}{f(a)-f(b)} = \\ &= a - \frac{f(a)(a-b)}{f(a)-f(b)} \end{aligned}$$

Таким чином, ми отримаємо перше наближення кореня вихідного рівняння $f(x)$, знаючи значення кінців інтервалу і значення ісходної функції $f(x)$ на кінцях інтервалу.

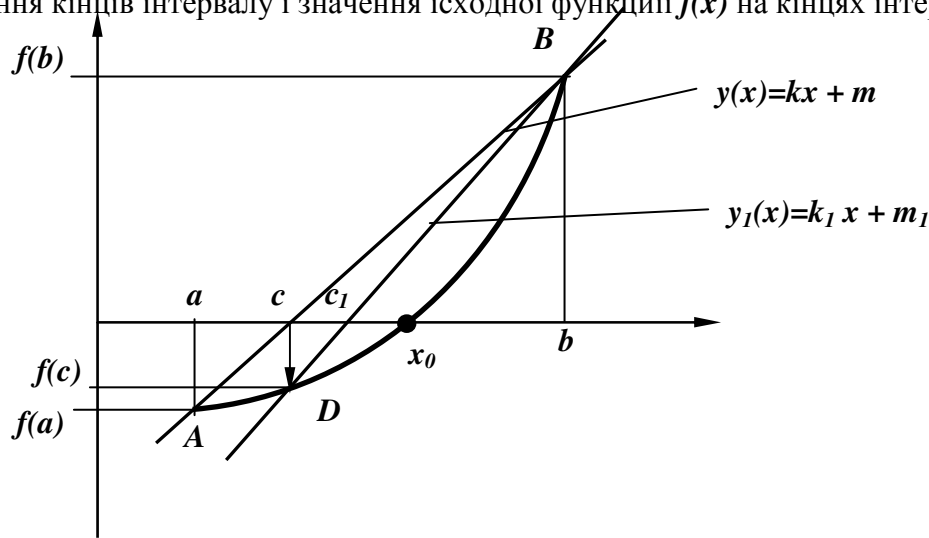


Рис.25 – Застосування методу хорд до графіку функції $f(x)$ в діапазоні x від a до b з визначенням наближених коренів у точках c і c_1

Подальше рішення рівняння відбувається ітераційним шляхом, другий крок якого зрозумілий з геометричної побудови на рисунку 25. Проводимо хорду через точки D і B . Точка D отримана при опусканні перпендикуляра з точки c на графік функції $f(x)$ до перетину. У цьому випадку хорда, яка проходить через точки D і B виражається лінійною залежністю $y_1(x)=k_1 x + m_1$. Повторивши аналогічні викладки, як і для попереднього випадка з перетином в точці c отримаємо:

$$\begin{aligned} \frac{f(c)-f(b)}{(c-b)} \cdot c_1 + f(c) - \frac{f(c)-f(b)}{(c-b)} \cdot c &= 0 \\ \frac{f(c)-f(b)}{(c-b)} \cdot c_1 &= \frac{f(c)-f(b)}{(c-b)} \cdot c - f(c) \\ c_1 &= \frac{f(c)-f(b)}{(c-b)} \cdot \frac{(c-b)}{f(c)-f(b)} \cdot c - \frac{f(c)(c-b)}{f(c)-f(b)} = \\ &= c - \frac{f(c)(c-b)}{f(c)-f(b)} \end{aligned}$$

Таким чином, використовуючи тільки значення лімітів інтервала відділення корення і значення функції в цих точках, ми отримаємо аналітичний вираз для визначення першого і наступного наближення корення вихідної функції (рівняня).

Важливо відзначити наступне:

- друге наближення корення отримано з використання першого наближення
- наступне наближення корення може бути отримано аналогічно другому і т.д.: тобто отриманий *алгоритм* обчислень наближеного значення корення.

Метод дотичних (метод Ньютона)

Математичний зміст похідної: Похідні показують для різних ділянок зміни функції, на скільки зростає функція при одному і тому ж збільшенні аргументу. Чи іншими словами, інтенсивність зростання (спадання) функції.

Якщо повторити ті ж міркування щодо зменшення відрізків прямих і відповідних інтервалів часу, то в межі ми зможемо з'ясувати, яка інтенсивність зміни функції від точки до точки.

$$u = \lim_{\Delta x \rightarrow 0} \frac{\Delta y}{\Delta x} = \frac{dy}{dx}$$

Графічно похідна має сенс дотичної, проведеної до графіка функції в даній точці.

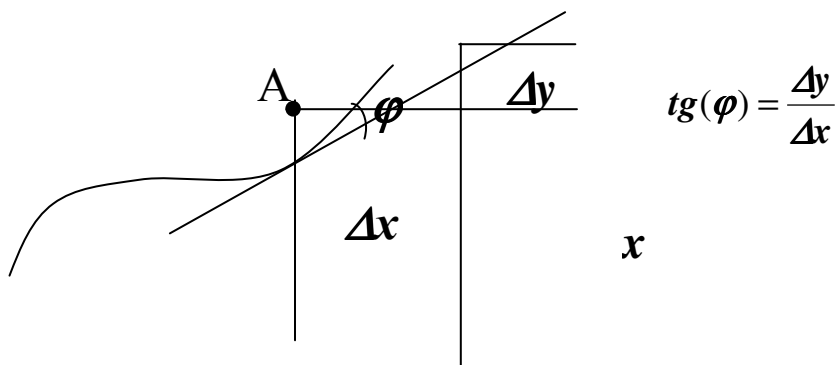


Рис.26 – Графічне представлення похідної у вигляді дотичної до графіка функції в даній точці

Таким чином, графічне зображення похідної в даній точці (саме в точці!) є дотичною до графіка функції (Рис. 26). А чисельне значення похідної, знову-таки в точці, є тангенс кута нахилу цієї дотичної.

Застосуємо поняття похідної, як характеристики інтенсивності зміни функції до чисельного розв'язання рівнянь.

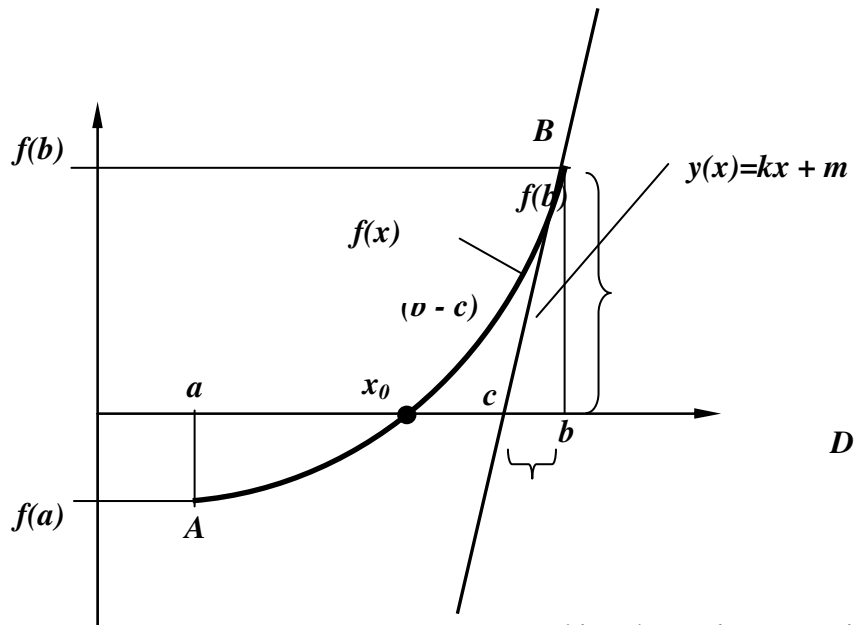


Рис.27 – Застосування методу дотичних до графіку функції $f(x)$ в діапазоні x від a до b

Нехай визначений інтервал відділення кореня. Ще раз нагадаємо, що ми просто обчислюємо значення функції, задаючи послідовно значення аргументу (наприклад, від -100 до +100). І на якомусь інтервалі $x_i = a$ і $x_{i+1} = b$ отримуємо значення функції $f(a) < 0$ і, відповідно, $f(b) > 0$. Це означає, що між обчисленими значеннями функція перетинає вісь x , тобто на цьому інтервалі знаходиться корінь рівняння. Проведемо до графіку функції $f(x)$ дотичну в точці B . Це ми робимо для того, щоб замінити криволінійний графік функції $f(x)$ прямолінійним відрізком. Рівняння цього відрізка прямої має простий вигляд (лінійне рівняння), а знаходження точки перетину цим графіком вісі x відомий, і дає точне значення кореня лінійного рівняння

Рівняння дотичної (лінійної залежності) буде мати вигляд:

$$\left. \frac{df(x)}{dx} \right|_{x=b} = f'(b) = \operatorname{tg}(\varphi) = \frac{f(b)}{(b-c)}$$

$$f'(b) \cdot (b-c) = f(b)$$

$$f'(b) \cdot b - f'(b) \cdot c = f(b)$$

$$f'(b) \cdot b - f(b) = f'(b) \cdot c$$

$$c = b - \frac{f(b)}{f'(b)}$$

Подібно описанному вище методу хорд ми отримаємо алгоритм рішення рівняння методом послідовних наближень. В отриманій формулі підраховується перше наближення корня, котре по суті змішує праву границю інтервалу виділення кореня ближче до шуканого значення x_0 . Наступна ітерація отримаємо в якості правої границі вже знайдене значення першого наближення c :

$$c_1 = c - \frac{f'(c)}{f(c)} \text{ і далі}$$

$$c_{i+1} = c_i - \frac{f'(c_i)}{f(c_i)}$$

Таким чином повторюючи підрахування по наведенній формулі отримаємо в кінці деяке значення c_1 таке, що є розв'язком (Рис. 28)

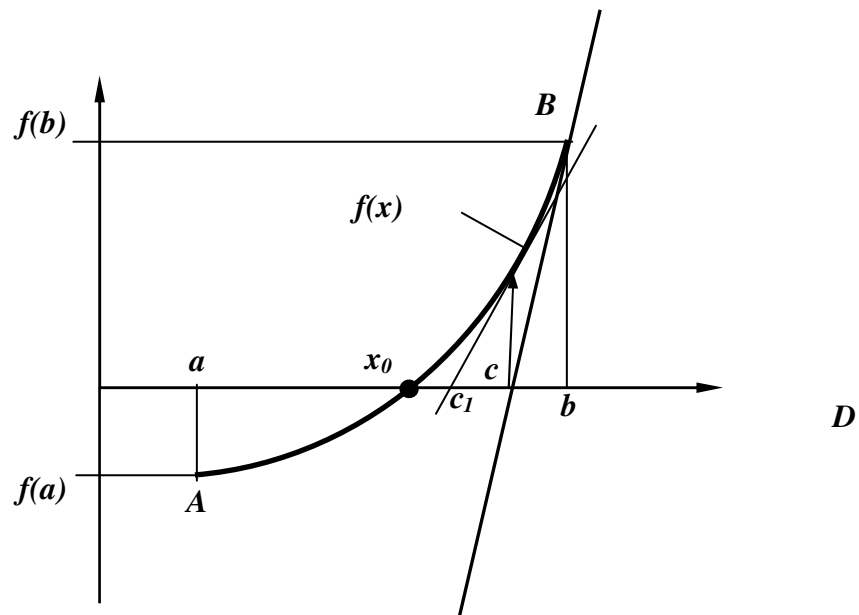


Рис.28 – Застосування методу дотичних до графіку функції $f(x)$ в діапазоні x від a до b з визначенням наближених коренів у точках c і c_1

Тоді ітераційні формули будуть виглядати, так:

а) 1-а ітерація: $c_1 = a - \frac{y(a)(b-a)}{y(b)-y(a)}$; 2-а ітерація: $c_2 = c_1 - \frac{y(c_1)(b-c_1)}{y(b)-y(c_1)}$

б) 1-а ітерація: $c_1 = b - \frac{y(b)(b-a)}{y(b)-y(a)}$; 2-а ітерація: $c_2 = c_1 - \frac{y(c_1)(c_1-a)}{y(c_1)-y(a)}$

и т.д. до значення c_i , котре буде відрізнятися від попереднього (c_{i-1}) на малу, наперед задану величину (наприклад, 0.01).

$$|c_{i+1} - c_i| < \epsilon,$$

где $\epsilon \ll 1$ наперед заданне мале число.

На цьому ітераційний процес закінчується.

Література: [осн. 1, 4, дод. 2].

Завдання на СРС: Вивчення матеріалу лекції. Самостійне опрацювання головних методів роботи в Mathcad: присвоєння змінних та констант; обчислення за допомогою формул. Розрахунок за формулами коренів рівнянь. Приклади для розв'язання рівнянь з курсу вищої математики. **Література:** [осн. 1, 4, дод. 2].

Лекція 10

Тема: Розв'язання рівнянь у MathCad з застосуванням вбудованих функцій

Mathcad підтримує ряд функцій для розв'язання рівнянь. Для цього в програмі Mathcad є так звані вбудовані функції, за допомогою яких можна знайти як чисельні, так і символні вирази для коренів рівняння.

Для розв'язання алгебраїчного рівняння з одним невідомим MathCAD має вбудовану функцію `root` (Рис. 29), яка в залежності від виду задачі може включати два або чотири аргументи, відповідно:

- $root(f(x), x);$
- $root(f(x), x, a, b),$ де
 - $f(x)$ — скалярна функція, яка визначає рівняння;
 - x — скалярна змінна, відносно якої розв'язується рівняння;
 - a, b — межі інтервалу, всередині якого відбувається пошук кореня.

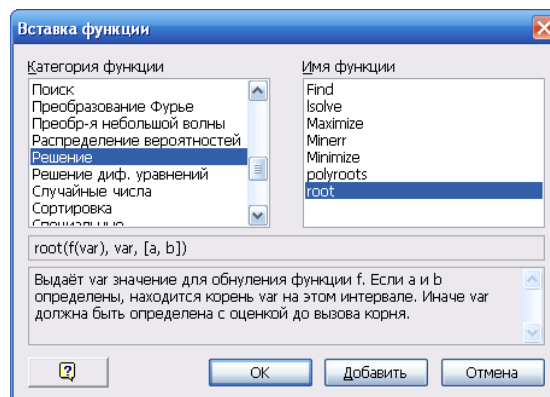


Рис. 29 - Виклик функцій `root` та `polyroots`

Розглянемо рішення та оптимізацію функцій і визначення коренів рівняння за допомогою функцій `root` та `polyroots`.

Рішення простих рівнянь за допомогою функції `root`

Функція `root` дозволяє розв'язувати прості рівняння з одним невідомим. Рішення рівняння може бути виконано наступним чином:

- задається заздалегідь перше наближення кореня;
- задається інтервал відділення кореня, тобто діапазон значень $[a, b]$ для x , в якому функція перетинає вісь x .

Для попереднього завдання першого наближення кореня як правило використовується побудова графіка функції та визначення інтервалу відокремлення коренів рівняння.

Функція `root` дає в результаті корінь рівняння, який звертає значення функції в нуль і знаходиться в інтервалі виділення кореня. Функція `root` «запускає» програму ітераційного обчислення кореня.

Завершення ітераційного процесу відбувається при досягненні значення кореня, яке відрізняється від отриманого на попередній ітерації на наперед задану малу величину. Цей процес характеризується точністю обчислень, яка задається з меню «Інструментів».

*Порядок відшукування кореня рівняння з одним невідомим за допомогою функції **root** з попереднім завданням першого наближення кореня (перший спосіб)*

Структура вбудованої функції **root** з попереднім завданням першого наближення кореня (перший спосіб):

root(<математичневизразфункції>,<ім'язмінної>)

1. Задається інтервал визначення функції.
2. Записується математичний вираз функції.
3. Будується графік функції і визначаються інтервали відділення кореня.
4. Задаються послідовно перші наближення всіх коренів і послідовно знаходяться їх наближені значення за допомогою функції **root**.

Нижче наведено приклад (Рис. 30) розв'язування рівняння 3-го ступеня, як це виконується в робочому вікні Mathcad.

Задаємо інтервал визначення функції: $x := -5, -4.99..5$ та записуємо математичний вираз функції: $y(x) := x^3 - 10 \cdot x + 2$.

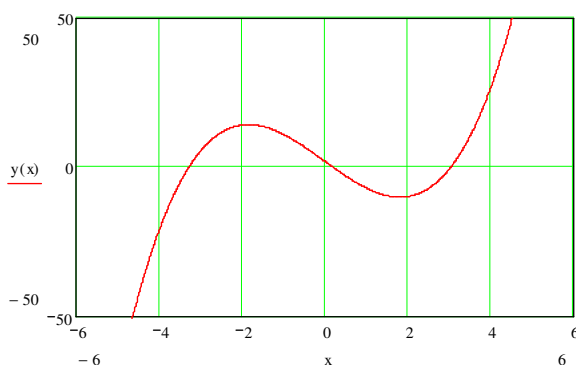


Рис. 30 - Приклад рішення рівняння за допомогою функції **root** з попереднім завданням першого наближення кореня (перший спосіб)

Визначаємо наближення коренів рівняння та обчислюємо їх точне значення за допомогою функції **root**:

$x := -5$ наближене значення першого кореня

$\text{root}(F(x), x) = -3.258$ точне значення першого кореня

$x := 0$ наближене значення другого кореня

$\text{root}(F(x), x) = 0.201$ точне значення другого кореня

$x := 3$ наближене значення третього кореня

$\text{root}(F(x), x) = 3.057$ точне значення третього кореня

*Порядок відшукування кореня рівняння з одним невідомим за допомогою функції **root** з попереднім завданням тільки інтервалів відділення кореня (другий спосіб)*

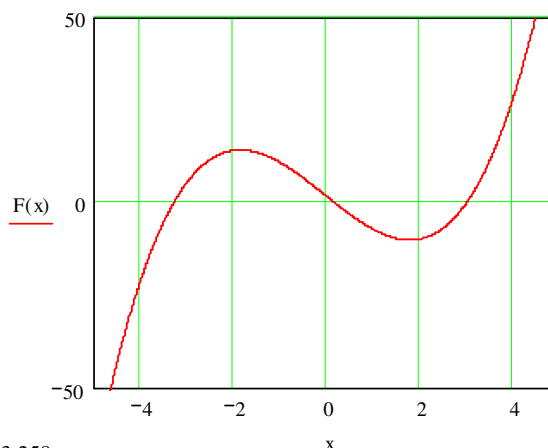
Структура вбудованої функції **root** з попереднім завданням тільки інтервалів відділення кореня (другий спосіб):

root(<математичневизразфункції>,<ім'язмінної>,<ліва межа інтервалу, права межа інтервалу>)

1. Задається інтервал визначення функції.
2. Записується математичне вираз функції.
3. Будується графік функції і визначаються інтервали відділення кореня.
4. Задаються послідовно інтервали відділення всіх коренів і послідовно знаходяться їх наближені значення за допомогою функції **root**.

Нижче наведено приклад (Рис.31) розв'язування рівняння 3-го ступеня, як це виконується в робочому вікні Mathcad.

$$F(x) := x^3 - 10x + 2$$



$$\text{root}(F(x), x, -5, -2) = -3.258$$

$$\text{root}(F(x), x, -2, 1) = 0.201$$

$$\text{root}(F(x), x, 1, 5) = 3.057$$

Рис.31 - Приклад рішення рівняння за допомогою функції root з попереднім завданням тільки інтервалів відділення кореня (другий спосіб)

*Рішення простих рівнянь за допомогою функції **polyroots***

Для визначення всіх коренів полінома використовується вбудована функція **polyroots**.

Нехай відомий вираз поліному:

$$a_n x^n + \dots + a_2 x^2 + a_1 x + a_0 = 0.$$

В цьому випадку можна використовувати функцію **polyroots**, яка не вимагає завдання попереднього завдання перше наближення кореня, а відразу дає в результаті всі значення коренів.

*Порядок відшукування коренів полінома за допомогою вбудованої функції **polyroots**:*

1. Коефіцієнти при невідомому і вільний член записуються у вигляді вектора-стовпця в порядку зростання ступеня при невідомому, починаючи з нульової.

$$\text{Коефіцієнти} := \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \dots \\ a_n \end{pmatrix}$$

2. Записується функція polyroots і виконується визначення вектора-стовпця:=.

Структура функції polyroots наступна:

polyroots(<ім'я вектора-стовпця з коефіцієнтами поліному>)

$$\text{polyroots}(\text{Coefficient}) := \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix}$$

Нижче наведено приклад (Рис.32) рішення рівняння 3-го ступеня, як це виконується в робочому вікні Mathcad.

$$F(x) := x^3 - 10 \cdot x + 2$$

$$\text{Coefficient} := \begin{pmatrix} 2 \\ -10 \\ 0 \\ 1 \end{pmatrix}$$

$$\text{polyroots}(\text{Coefficient}) = \begin{pmatrix} -3.258 \\ 0.201 \\ 3.057 \end{pmatrix}$$

Рис.32 - Приклад рішення рівняння за допомогою функції **polyroots**

Слід особливо відзначити, що функції **root** і **polyroots** дозволяють вирішувати ОДНЕ РІВНЯННЯ З ОДНИМ НЕВІДОМИМ і завжди дають чисельний результат коренів.

Рішення складних рівнянь за допомогою функції root

Нехай задано складне рівняння. У цьому випадку, для визначення коренів рівняння можна застосувати відомий прийом поділу даного рівняння на дві незалежні функції та побудувати їх графіки.

Дана операція в робочому вікні Mathcad буде виглядати як на рисунку 33:

$$\sin(x) = \frac{x+1}{10}$$

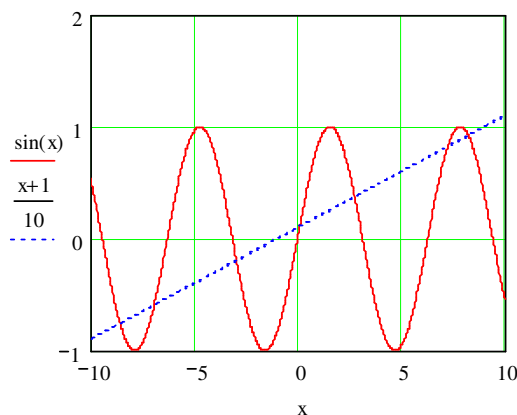


Рис.33 - Приклад рішення складного рівняння за допомогою функції **root**

Після попереднього визначення коренів рівняння записується рівняння у вигляді знак рівності з меню булевих операцій означає не рівність, як таке, а лише оцінку рівності нулю даного виразу.

Рішення рівняння буде мати вигляд:

$$x := 9$$

$$\text{root}\left(\sin(x) - \frac{x+1}{10}, x\right) = 8.245$$

$$x := 7$$

$$\text{root}\left(\sin(x) - \frac{x+1}{10}, x\right) = 7.254$$

$$x := 3$$

$$\text{root}\left(\sin(x) - \frac{x+1}{10}, x\right) = 2.756$$

$$x := 0$$

$$\text{root}\left(\sin(x) - \frac{x+1}{10}, x\right) = 0.111$$

Процедура пошуку рішень аналогічна описаній вище для послідовного відшукування всіх коренів рівняння.

Рішення систем рівнянь за допомогою вирішального блоку SolveBlock

Вирішальний блок **SolveBlock** може використовуватися для вирішення будь-яких систем рівнянь з досить великою кількістю рівнянь та відповідних невідомих.

Порядок вирішення систем рівнянь за допомогою вирішального блоку SolveBlock наступний:

1. Визначити перші наближення коренів.
2. Набрати слово **given** (дано), що означає в перекладі як символ «дано ці рівняння». Слово **given** визначає поле розв'язання рівнянь в межах цього поля виконуються дії.
3. Записати рівняння з використанням булевої операції оцінки рівності нулю рівняння (*жирні лінії знака рівності*). Для цього можна використовувати меню булевих операцій або команду одночасно натиснутих клавіш [Ctrl] і [=].
4. Нижче набрати текст **Find (x,y,z,...) =** . Вміст команди **Find** вказати всі невідомі системи.

$$3 \cdot x + 4 \cdot y - 8 \cdot z = 3 \quad -5 \cdot x + 5 \cdot y - 9 \cdot z = 4 \quad 7 \cdot x + 4 \cdot y - 2 \cdot z = 0$$

$$x := 1 \quad y := -1 \quad z := 1$$

Given

$$3 \cdot x + 4 \cdot y - 2 \cdot z - 5 = 0$$

$$-5 \cdot x + 5 \cdot y - 4 \cdot z - 4 = 0$$

$$7 \cdot x + 4 \cdot y - z = 0$$

$$\text{Find}(x, y, z) = \begin{pmatrix} 3.545 \\ -11 \\ -19.182 \end{pmatrix}$$

У блоці використовується одна з наступних двох функцій:

find(v1, v2, ..., vn) — повертає значення однієї або кількох змінних для точного рішення;

minerr(v1, v2, ..., vn) — повертає значення однієї або кількох змінних для наближеного рішення. Між цими функціями існують принципові відмінності.

Перша функція **find(v1, v2, ..., vn)** використовується, коли рішення реально існує (хоча і не є аналітичним).

Друга функція **minerr(v1, v2, ..., vn)** намагається знайти максимальне наближення навіть до неіснуючого рішення шляхом мінімізації середньоквадратичної похибки рішення.

З допомогою цих функцій можна вирішувати і поодинокі рівняння, наприклад, нелінійні.

Література: [осн. 1, 4, дод. 2].

Завдання на СРС: Вивчення матеріалу лекції. Самостійне опрацювання розв'язання рівнянь за допомогою вбудованих функцій Mathcad: find, minerr. **Література:** [осн. 1, 4, дод. 2].

Лекція №11

Тема: Операції з матрицями та векторами. Ранжовані змінні у MathCAD

Ранжировані змінні — особливий клас змінних, який у системі MathCAD найчастіше заміняє цикли (однак повноцінною така заміна не є). Ці змінні мають ряд фіксованих значень, мінливих з певним кроком. Ранжировані змінні характеризуються ім'ям та індексом кожного свого елемента. Ранжировані змінні широко застосовуються для представлення числових значень функцій у вигляді таблиць, а також для побудови їх графіків. Будь-який вираз з ранжированими змінними після знака рівності ініціює таблицю виводу (Рис.34).

Важливим типом даних у системі MathCAD є масиви. Масив — має унікальне ім'я сукупність кінцевого числа числових чи символьних елементів, впорядкованих заданим чином і мають певні адреси. У системі MathCAD використовуються масиви двох найбільш поширених типів: одномірні (вектори) та двовимірні (матриці). Вектори можуть бути двох типів: вектори-рядки і вектори-стовпці.

Порядковий номер елемента, що є його адресою, називається індексом. Індеси можуть мати тільки цілочисельні значення. Нижня межа індексації задається значенням системної змінної **ORIGIN**, яка може приймати значення 0 або 1. Елементи матриць є індексованими змінними, імена яких збігаються з іменами матриць. Але в цьому випадку для кожної індексованої змінної зазначаються два індекси: один — для номера рядка, другий — номер стовпця.

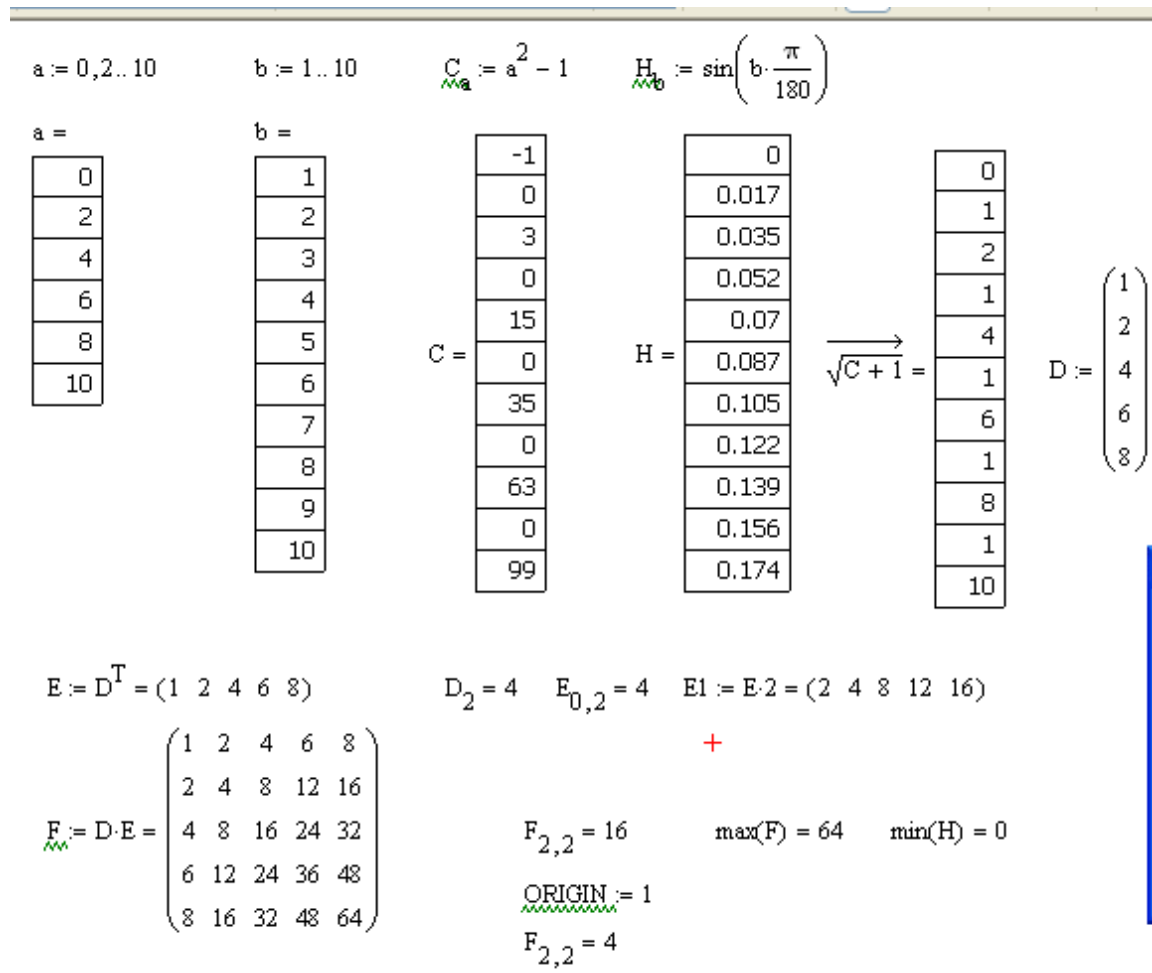


Рис. 34 – Приклади виразів з ранжированими змінними у MathCAD

Для роботи з векторами та матрицями система MathCAD містить ряд операторів і функцій: арифметичні вирази, множення на матрицю або скаляр, транспонування, обчислення визначника, сортування матриці або окремих рядків, стовпців. На рисунку 35 наведено приклади завдання матриць через коефіцієнти (матриці A і B) та множення між ними з перевіркою через оператор $lsolve(A,B)$ та завдання матриці через вираз зі змінними $M(i, j)$ і графічне її зображення у MathCAD.

$$A := \begin{pmatrix} 2 & 4 & 6 \\ 1 & -4 & -5 \\ 0 & 7 & -8 \end{pmatrix} \quad B := \begin{pmatrix} 4 \\ -5 \\ 12 \end{pmatrix}$$

$$X := A^{-1} \cdot B = \begin{pmatrix} -0.26 \\ 1.462 \\ -0.221 \end{pmatrix}$$

$$X1 := \text{lsolve}(A, B) = \begin{pmatrix} -0.26 \\ 1.462 \\ -0.221 \end{pmatrix}$$

задание матрицы

$$i := 1..4 \quad j := 1..4 \quad M_{(i,j)} := \frac{i^2}{2} - \frac{j^3}{8}$$

$$M = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0.375 & -0.5 & -2.875 & -7.5 \\ 0 & 1.875 & 1 & -1.375 & -6 \\ 0 & 4.375 & 3.5 & 1.125 & -3.5 \\ 0 & 7.875 & 7 & 4.625 & 0 \end{pmatrix}$$

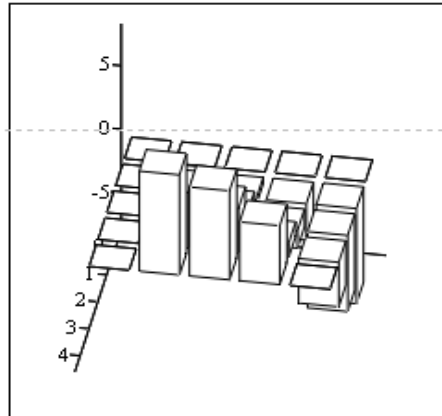


Рис. 35 – Приклади завдання матриць через коефіцієнти (матриці A і B) та через вираз зі змінними для матриці $M(i, j)$

Література: [осн. 1, 4, дод. 2].

Лекція 12

Тема: Побудова графіків, графічне зображення у MathCad матриць, тривимірні графіки

Для побудови графіків використовуються шаблони. Для найбільш поширених графіків в декартовій та полярній системах координат у MathCAD передбачений дуже зручний спосіб побудови графіків без попереднього завдання функції та ранжирування змінної.

Незаповнений шаблон графіка являє собою великий порожній прямокутник з шаблонами даних у вигляді темних маленьких прямокутників, розташованих близько осей абсцис і ординат майбутнього графіка. У середині шаблону даних треба помістити ім'я змінної (x у осі абсцис) і задати ім'я або формули функцій (y в осі ординат). Прості функції, якщо вони в подальшому не використовуються, можна вказати шаблон самого графіка. Якщо будуються графіки кількох функцій в одному шаблоні, то для їх поділу слід використовувати коми. При зазначеному спрощеному способі побудови графіків вони будуються при зміні незалежної змінної від -10 до +10 (рис.36). Надалі операцією форматування цей діапазон і всі інші дані графіків можна змінити. Щоб відбулося побудова графіка в автоматичному режимі обчислень, досить вивести курсор за межі графічного об'єкта.

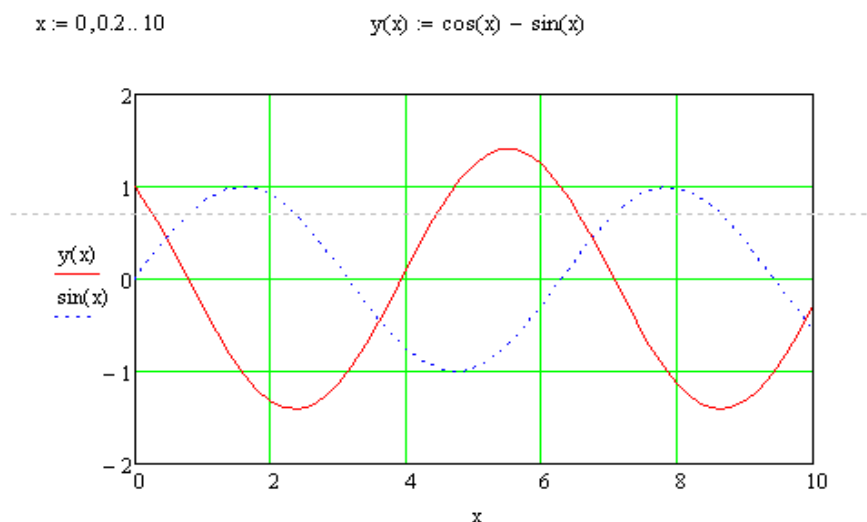


Рис. 36 - Графік функції $y=f(x)$ від -10 до+ 10

Параметри графіків можна змінювати подвійним клацанням миші.

Можна виставити лінії сітки, тип і колір ліній, підписи осей і графіка (рис. 37).

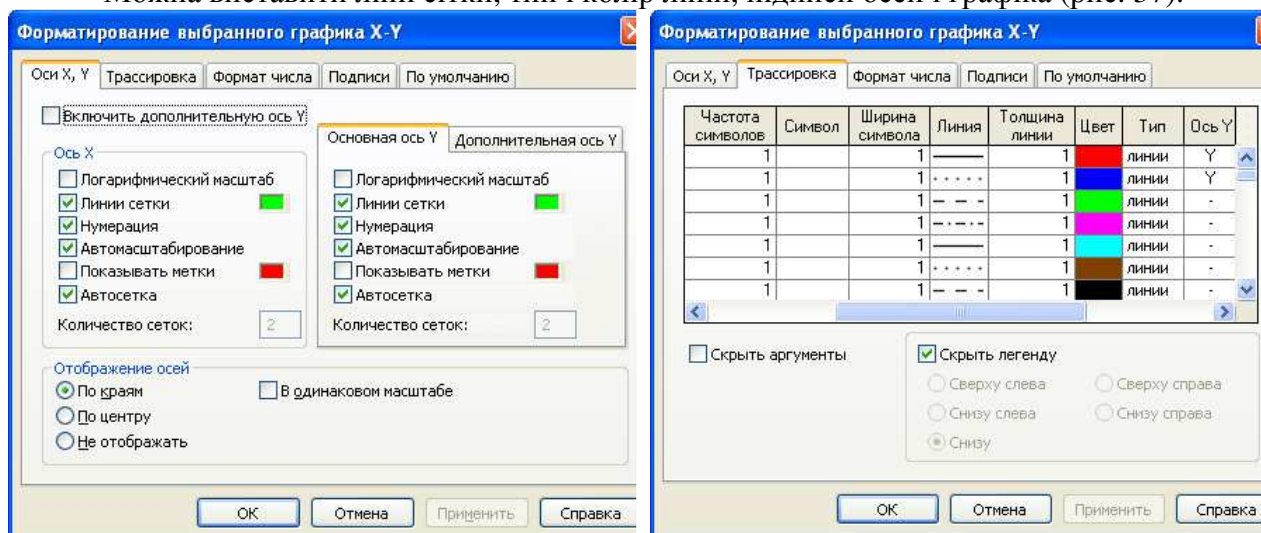


Рис. 37 - Форматування параметрів графіків у MathCAD

Графік функції $y=f(x)$ від -10 до+ 10 після форматування представлений на рис. 38.

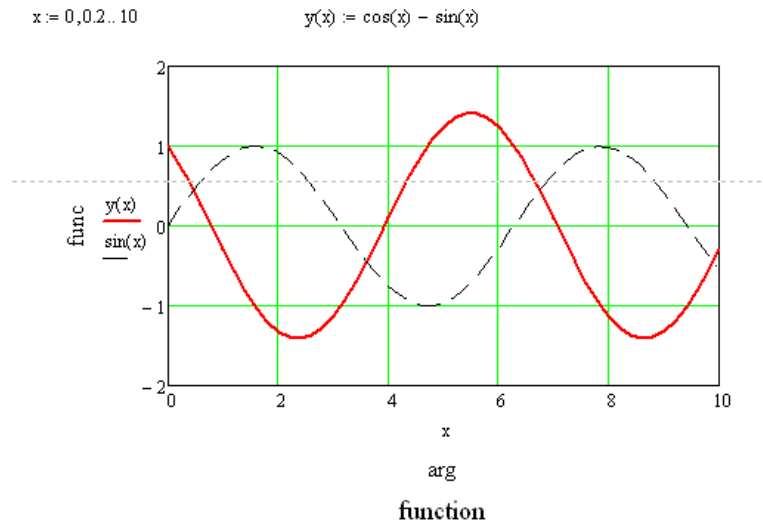


Рис. 38 - Графік функції $y=f(x)$ від -10 до+ 10 після форматування

MathCAD дозволяє створювати 3-х мірні графіки у вигляді поверхні, набору точок, векторів, стовпців, ізоліній. Дані попередньо повинні бути представлені матрицею або функцією 2-х змінних (Рис. 39).

$$f(x, y) := 3 \cdot x^2 - \frac{y^3}{3}$$

$x := 1 \dots 10$ $y := 1 \dots 10$

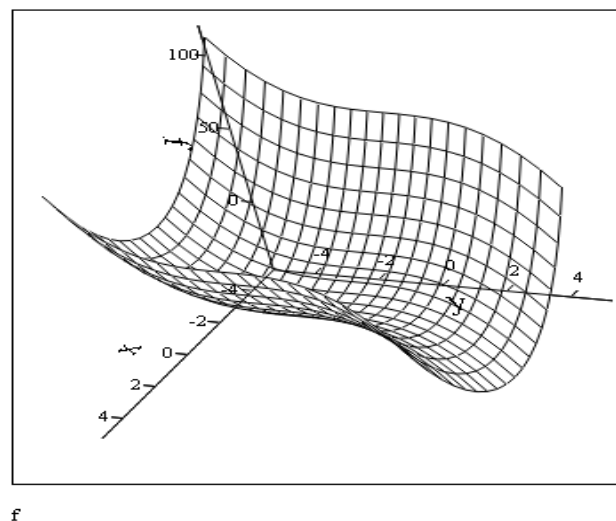


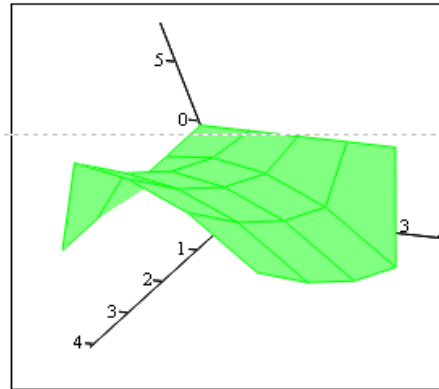
Рис. 39 - Побудування 3-х мірного графіку у MathCAD

У властивостях графіка можна змінювати кут повороту і нахилу осей, вид відображення і т.п. Корисна функція «приховати невидимі лінії», що полегшує сприйняття. Це показано на прикладі завдання матриці M у MathCAD та побудові графіку матриці M (Рис. 40).

задание матрицы

$$i := 1..4 \quad j := 1..4 \quad M_{(i,j)} := \frac{i^2}{2} - \frac{j^3}{8}$$

$$M = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0.375 & -0.5 & -2.875 & -7.5 \\ 0 & 1.875 & 1 & -1.375 & -6 \\ 0 & 4.375 & 3.5 & 1.125 & -3.5 \\ 0 & 7.875 & 7 & 4.625 & 0 \end{pmatrix}$$



M

Рис. 40 - Завдання матриці у MathCAD і побудова графіку матриці M

Натисканням лівої клавіші миші на графік у 3-х мірному просторі у MathCAD викликається вікно «Формат 3D графика», де є можливість завдання параметрів форматування графіків у 3-х мірному просторі у MathCAD (рис.41).

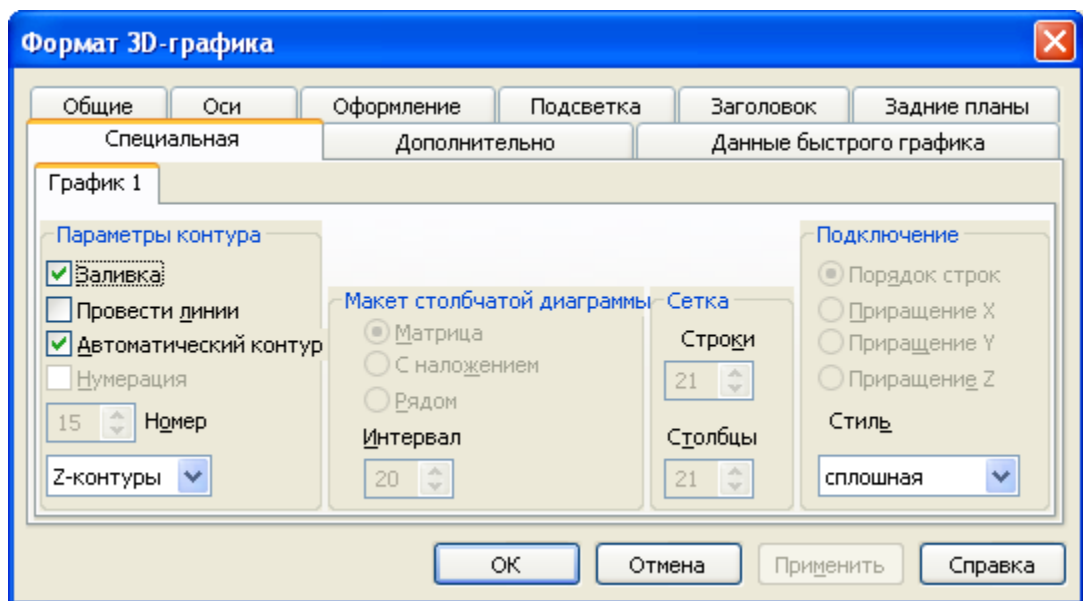


Рис. 41 - Форматування параметрів графіку у MathCAD

Підвищення наочності сприйняття 3D-поверхонь забезпечується застосуванням функціонального забарвлення. Вона дає додаткову інформацію про третьому вимірі: чим нижче розташована поверхню фігури, тим вона темніше (Рис. 42).

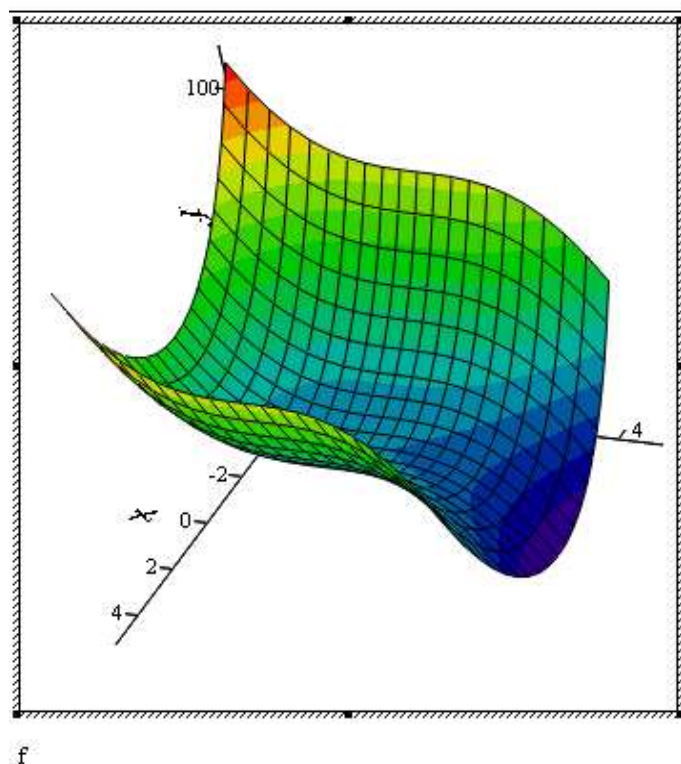


Рис. 42 - Побудування 3-х мірного графіку у MathCAD з забарвленням

Оскільки графік будується на основі матриці, яка містить координати вершин фігури, то справжні масштаби по осях X і Y не відомі і на малюнках не проставляються. Можуть виводитися просто порядкові номери елементів матриць в заданому напрямку (X і Y).

Література: [осн. 1, 4, дод. 2].

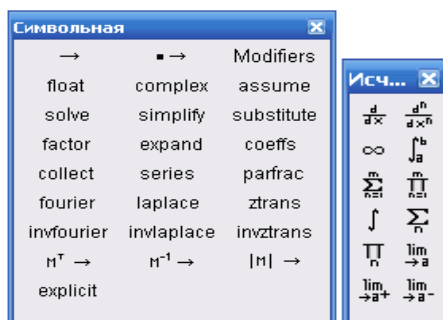
Завдання на СРС: Вивчення матеріалу лекції. Самостійне виконання побудови графіків різних видів та їх редагування. **Література:** [осн. 1, 4, дод. 2].

Лекція 13

Тема: Символьні обчислення MathCad. Символьне розв'язання лінійних та нелінійних рівнянь

Mathcad – програма, яка дозволяє виконувати так названі символьні математичні операції. Коли Mathcad обчислює символьно, результатом є інший вираз.

Для символьних обчислень використовуються дві панелі: **Символьний (Symbolic)** (Рис.43 (а)) і **Обчислення (Calculus)** (Рис.43 (б)).



a)

б)

Рис. 43- Меню Symbolic (a); Calculus (б) на панелі інструментів

Для проведення символічних операцій потрібно спочатку виділити об'єкт, над якими ці операції будуть виконуватися. Об'єктом операції може бути математичне вираз, частина виразу або заданої користувачем функції, результат попередньої операції і т.д. Для виділення деякої змінної в об'єкті потрібно підвести до її кінця курсор миші і натиснути ліву клавішу. Змінна буде відзначена жирною синьою лінією, розташованою відразу після змінної. Переміщуючи курсор по полю об'єкту і натискаючи ліву клавішу повторно, можна виділити окремі частини виразу або вираз в цілому.

Операції з виділеними виразами

До операцій з виділеними виразами відносяться:

Evaluate (Обчислити) — перетворити вираз з вибором виду перетворень з підменю.

Evaluate Symbolically [Shift-F9] (Обчислити) забезпечує роботу з математичними виразами, що містять вбудовані в систему функції і представленими в різному виді: поліноміальному, у вигляді сум і добутків, похідних і інтегралів і т. д. Операція прагне зробити всі можливі чисельні обчислення і уявити вираз в найбільш простому вигляді.

Simplify (Спростити) — спростити виділений вираз, що містять алгебраїчні та тригонометричні функції, а також многочленів з виконанням таких операцій, як скорочення подібних доданків, приведення до спільного знаменника, використання основних тригонометричних тотожностей і т.д, (наприклад, $\sin(x)^2 + \cos(x)^2 \text{ Simplify} \rightarrow 1$);

Expand (Розкласти) — розкрити вираз, наприклад, за ступенями:

$(X + Y)(X - Y)$ одержуємо $X^2 - Y^2$,

Factor (Розкласти) — розкласти число вираз на множники, наприклад: $X^2 - Y^2$ дасть $(X + Y)(X - Y)$,

Collect (Розкласти) — зібрати доданки, подібні до виділеного за подвыражению) виразу, що може бути окремою змінною чи функцією зі своїм аргументом (результатом буде вираз, полиномиальное щодо обраного виразу).

Операції з виділеними змінними

До числа операцій з виділеними змінними відносяться

Solve (Вирішити) — знайти значення виділеної змінної, відносно змінної) при яких містить її вираз стає рівним нулю (розв'язати рівняння або нерівність щодо виділеної змінної);

Differentiate — диференціювати весь вираз, що містить виділену змінну, по відношенню до цієї змінної (інші змінні розглядаються як константи);

Задамо коефіцієнти нелінійного рівняння:

$a := 4, b := 28, c := -14, d := -124$.

Запишемо нелінійне рівняння функції: $y(x) := a \cdot x^3 + b \cdot x^2 + c \cdot x + d$.

Користуючись меню **Calculus**, викликаємо піктограму диференціювання $\frac{d}{d_}$. У плейс-холдери отриманого шаблону диференціювання вписуємо функцію $y(x)$ та ім'я змінної x і виконуємо символічне диференціювання натисканням піктограми символічних обчислень з меню **Symbolic** " \rightarrow ", або [Ctrl][.].

На екрані отримуємо: $\frac{d}{dx} y(x) \rightarrow 12 \cdot x^2 + 56 \cdot x - 14$

Для обчислення похідної іншого порядку з меню **Calculus** викликати піктограму похідної вищих порядків, відповідно заповнюємо плейс-холдери і обчислюємо другу похідну: $\frac{d^2}{dx^2} y(x) \rightarrow 24 \cdot x + 56$

Для візуалізації результатів обчислень і розуміння суті виконаних операцій побудуємо графіки всіх функцій, отриманих при диференціюванні, при цьому для зручності побудови графіка позначимо першу та другу похідні як функції $y_1(x)$ та $y_2(x)$:

$$y_1(x) := \frac{d}{dx} y(x); y_2(x) := \frac{d^2}{dx^2} y(x).$$

Побудуємо графік, вписуючи до плейс-холдерів імена змінної x , функцій $y(x)$, $y_1(x)$, $y_2(x)$ (відокремлюючи їх комою), а також вказуючи масштаб по абсцисі x та ординаті y (Рис.44).

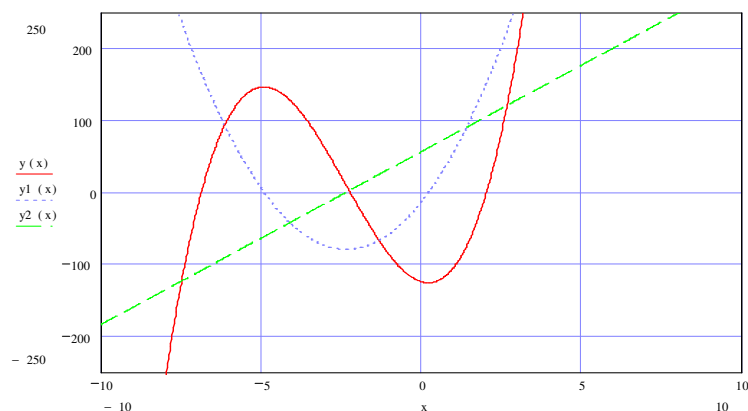


Рис. 44 - Вид графіка функції $y(x) := 4 \cdot x^3 + 28 \cdot x^2 - 14 \cdot x - 124$ та математичних виразів першої і другої похідних

Для обчислення границь використовуються ті самі меню, що і для похідних: **Symbolic** і **Calculus**.

Визначимо коефіцієнти функції та саму функцію:

$$a := 2; y(x) := \frac{a \cdot \sin(x)}{x}.$$

Користуючись меню **Calculus**, визначаємо границі при $x \rightarrow 0^-$, $x \rightarrow 0^+$, $x \rightarrow 0^\infty$:

$$\lim_{x \rightarrow 0^-} y(x) \rightarrow 2; \lim_{x \rightarrow 0^+} y(x) \rightarrow 2; \lim_{x \rightarrow \infty} y(x) \rightarrow 0.$$

Для візуалізації операцій обчислення границь побудуємо графік функції і впевнимися в тому, що знайдені границі відповідають значенням функції у вказаних межах значень змінної (Рис.45).

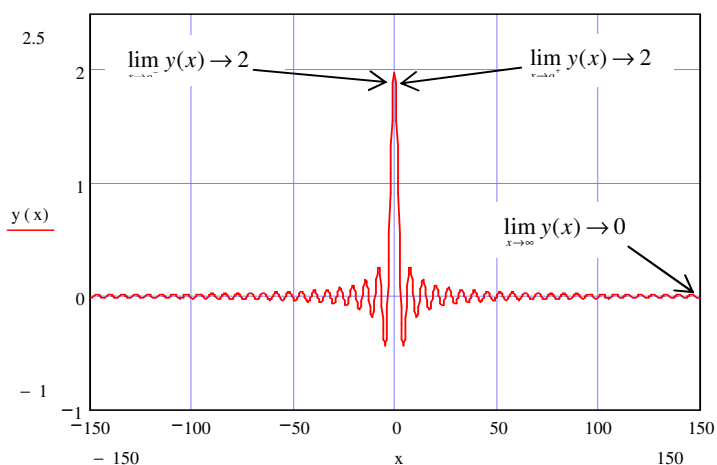


Рис. 45 - Графік функції $y(x) := \frac{2 \cdot \sin(x)}{x}$ та позначення знайдених границь

Розкладення функції в ряд виконується за допомогою команди **series** в меню **Symbolic**:

<ім'я функції> **series**, <ім'я змінної>, порядок розкладення →

Визначимо функцію: $y(x) := \sin(x)$.

Розкладемо функцію в ряд з порядком розкладення 4 та 11 відповідно:

$$F(x) \text{series}, x, 4 \rightarrow 1 \cdot x - \frac{1}{6} \cdot x^3;$$

$$F(x) \text{series}, x, 11 \rightarrow 1 \cdot x - \frac{1}{6} \cdot x^3 + \frac{1}{120} \cdot x^5 - \frac{1}{5040} \cdot x^7 + \frac{1}{362880} \cdot x^9.$$

Для зручності побудови графіка позначимо функції розкладення в ряд з порядком розкладення 4 та 11 як функції $y_1(x)$ та $y_2(x)$ відповідно:

$$F_1(x) := 1 \cdot x - \frac{1}{6} \cdot x^3; \quad F_2(x) := 1 \cdot x - \frac{1}{6} \cdot x^3 + \frac{1}{120} \cdot x^5 - \frac{1}{5040} \cdot x^7 + \frac{1}{362880} \cdot x^9.$$

Побудуємо графіки оригінальної функції $y(x)$ та поліномів $y_1(x)$ і $y_2(x)$ (Рис.46).

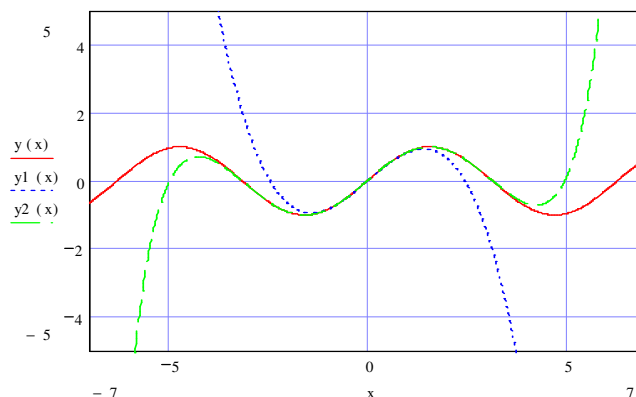


Рис. 46 - Графіки функції $y(x) := \sin(x)$ та поліномів з порядком розкладення 4 та 11

З рис. 46 бачимо, що поліном 11-го більш високого порядку замінює оригінальну функцію в більшому діапазоні змінної x , ніж поліном 4-го порядку.

Integrate (Інтегрувати) — інтегрувати весь вираз, що містить виділену змінну, по цій змінній.

Оператор символьних обчислень " \rightarrow " сам по собі є потужним інструментом, який дозволяє виробляти різні дії з виразами.

$$\int_a^b x^2 dx \rightarrow \frac{1}{3} \cdot b^3 - \frac{1}{3} \cdot a^3$$

$$x := 8$$

$$y + 2 \cdot x \rightarrow y + 16$$

$$y^2 \rightarrow y^2$$

$$\sqrt{17} \rightarrow 17^{\frac{1}{2}} = 4.123$$

Вьчислити

$$2 \cdot x^2 - 8 \cdot x + 5 = 0 \text{ solve, } x \rightarrow \begin{pmatrix} \frac{\sqrt{6}}{2} + 2 \\ 2 - \frac{\sqrt{6}}{2} \end{pmatrix}$$

$$2 \cdot x^2 - 8 \cdot x + 5$$

Дифференцировать

$$4 \cdot x - 8$$

Интегрировать

$$\frac{2 \cdot x^3}{3} - 4 \cdot x^2 + 5 \cdot x$$

Література: [осн. 1, 4, дод. 2].

Завдання на СРС: Вивчення матеріалу лекції. Самостійне виконання символьних обчислень за завданнями з вищої математики. **Література:** [осн. 1, 4, дод. 2].

Лекція 14

Тема: Елементи автоматичних обчислень у MathCad. Програмування математичних обчислень на вбудованій мові MathCad

Можливості програмування зосереджені в складальній панелі програмних елементів, показаної на рис. 47.

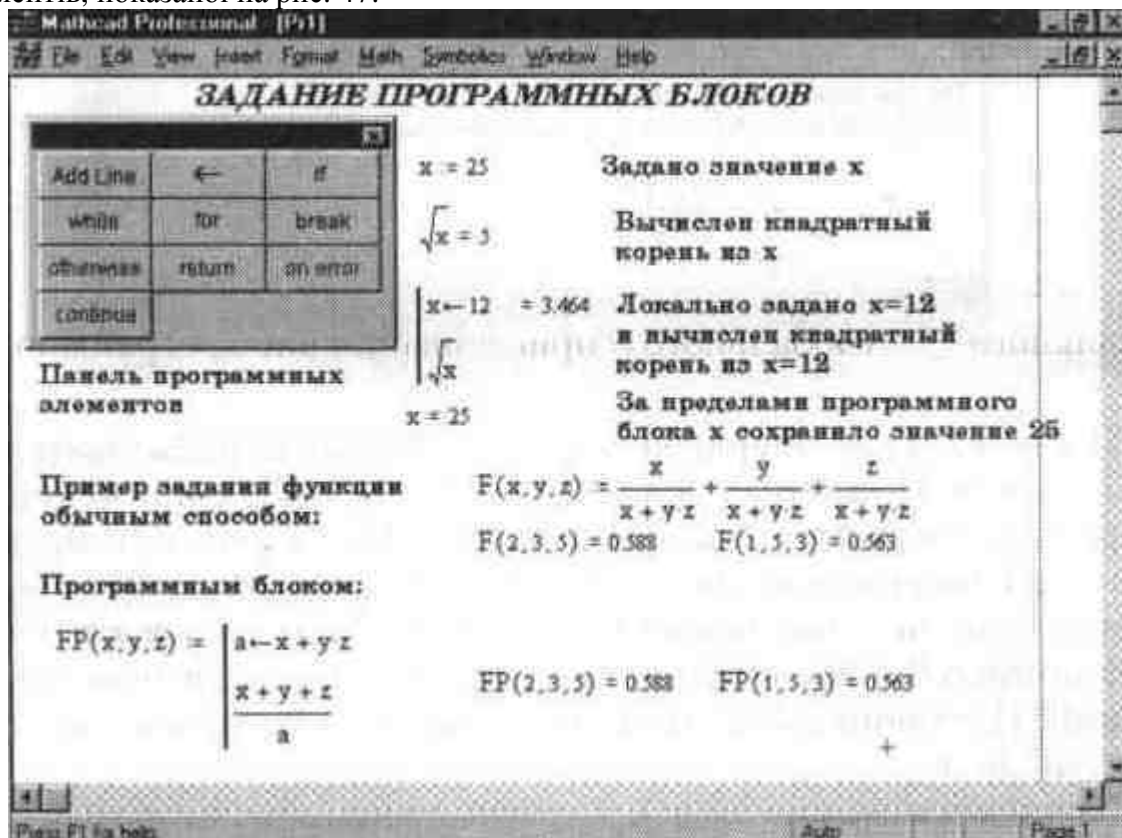


Рис. 47 – Завдання програмних блоків

Як видно з рис. 47, програмний модуль в системі MathCAD перетворився на самостійний модуль, що виділяється в тексті документа жирною вертикальною рисою. Він може вести себе як безіменна функція, яка не має параметрів, але повертає результат - перший приклад на обчислення квадратного кореня з числа 12. Програмний модуль може виконувати і роль тіла функції користувача з ім'ям і параметрами - другий приклад.

Огляд програмних операторів в системі MathCAD:

AddLine - створює і при необхідності розширює жирну вертикальну лінію, праворуч від якої в шаблонах задається запис програмного блоку;

<- - Символ локального привласнення (в тілі модуля);

if - оператор умовного вираження;

for - оператор завдання циклу з фіксованим числом повторення;

while - оператор завдання циклу типу "поки" (цикл виконується, поки виконується деяка умова);

otherwise - оператор іншого вибору (зазвичай застосовується з if);

break - оператор переривання;

continue - оператор продовження;

return - оператор повернення;

on error - оператор обробки помилок.

Оператор **AddLine** виконує функції розширення програмного блоку. Розширення фіксується подовженням вертикальної риси програмних блоків або їх деревовидним розширенням. Завдяки цьому в принципі можна створювати скільки завгодно великі програми.

Оператор <- виконує функції внутрішнього локального привласнення. Наприклад, вираз $x \leftarrow 123$ привласнює змінній x значення 123. Локальний характер присвоювання означає, що таке значення x зберігає тільки в тілі програми. За межами тіла програми значення змінної x може бути невизначеним або рівним значенню, яке задається операторами локального: $=$ і глобального $=$ присвоювання поза програмного блоку.

Оператор **if** є оператором умовного виразу. Він задається у вигляді:

Вираз if Умова

Якщо Умова виконується, то повертається значення Вирази. Спільно з цим оператором часто використовуються оператор переривання **break** і оператор іншого вибору **otherwise**.

Оператор **for** служить для організації циклів з заданим числом повторень. Він записується у вигляді:

For Var e Nmin .. Nmax

Цей запис означає, що якщо змінна **Var** змінюється з кроком + 1 відзначення **Nmin** до значення **Nmax**, то вираз, поміщений в шаблон, виконуватиметься. Змінну лічильника **Var** можна використовувати у виразах програми.

Оператор **while** служить для організації циклів, щодіють до тих пір, поки виконується деяка Умова. Цей оператор записується у вигляді:

while Умова •

Виконуваний вираз записується на місце шаблону.

Оператор іншого вибору **otherwise** зазвичай використовується спільно з оператором **if**.

Це пояснює наступна програмна конструкція:

$f(x) := 1$ if $x > 0$ повертає 1, якщо $x > 0$
-1 Otherwise повертає -1 у всіх інших випадках

Оператор **break** викликає переривання роботи програми всякий раз, коли він зустрічається. Найчастіше він використовується спільно з оператором умовного вираження **if** і операторами циклів **while** і **for**, забезпечуючи перехід в кінець тіла циклу.

Оператор **continue** використовується для продовження роботи після переривання програми. Зазвичай він застосовується спільно з операторами завдання циклів **while** і **for**, забезпечуючи після переривання назад до початку циклу.

Оператор **return** перериває виконання програми і повертає значення свого операнда, що стоїть слідом за ним. Наприклад, у наведеному нижче випадку

return 0 if $x < 0$

повертатиметься значення 0 при будь-якому $x < 0$.

Оператор **onerror** є оператором обробки помилок, що дозволяє створювати конструкції обробників помилок. Цей оператор задається у вигляді:

Вираз_1 onerror Вираження_2

Якщо при виконанні *Вираз_1* виникає помилка, то виконується *Вираз_2*. З оператором **onerror** пов'язана функція **error**, яка зазвичай використовується для повернення текстового повідомлення про помилку.

Приклади програмування

Незважаючи на настільки скромний набір програмних засобів, вони дають системі MathCAD саме ті можливості, які раніше попросту були відсутні: завдання функцій з апаратом локальних змінних, завдання різних видів циклів (у тому числі вкладених), спрощення алгоритмів за допомогою операцій присвоювання та реалізація за класичними алгоритмами ітераційних і рекурсивних процедур. На рис. 48 показана частина цих можливостей.

Зверніть особливу увагу на другий приклад обчислення факторіала. Тут використано завдання одного програмного модуля всередині іншого. Взагалі кажучи, для кількох підмодулів, які повинні виконуватися в складі циклів, треба використовувати їх об'єднання у вигляді жирної вертикальної риси. Для цього, як і для вихідного завдання блоку, служить команда AddLine, яка додає в програмний модуль додаткову вертикальну риску для підмодуля.

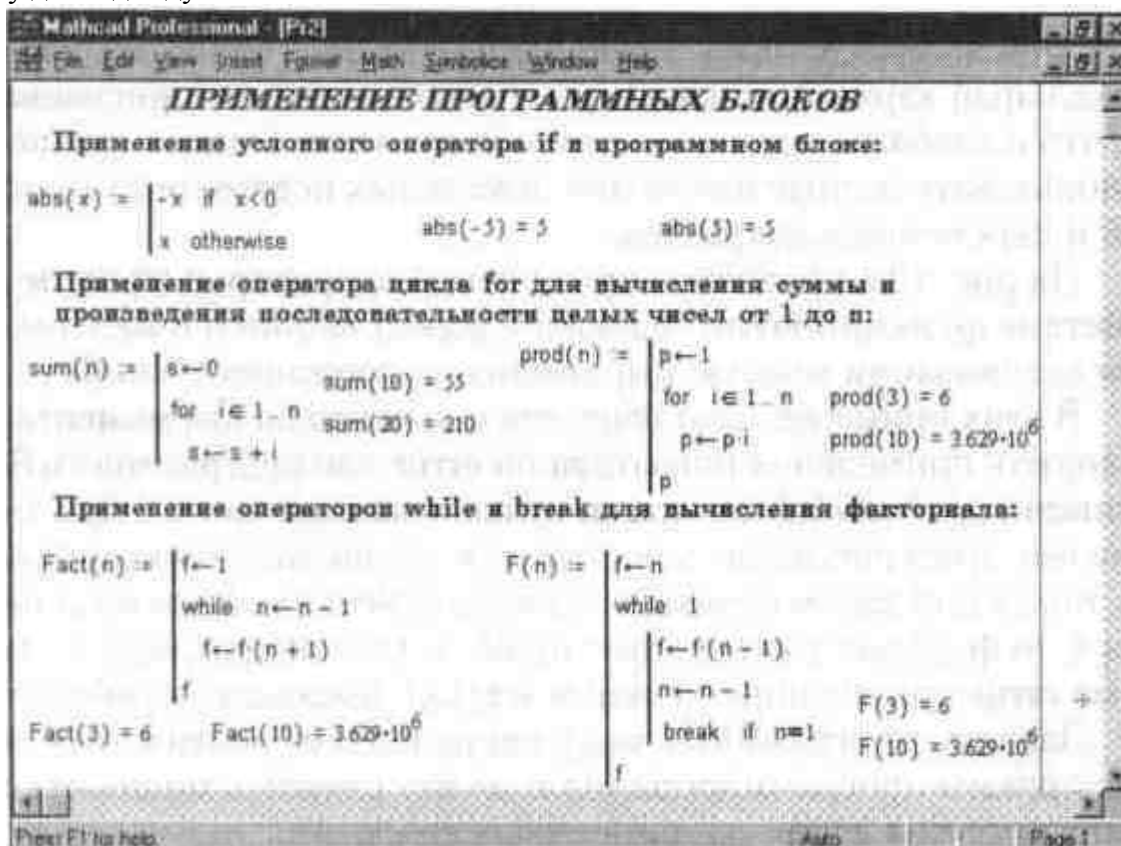


Рис. 48 - Приклади завдання програмних блоків в системі MathCAD

Програмний модуль по суті є функцією, але описаної із застосуванням згаданих суто програмних засобів. Ця функція може повертати значення, яке визначається останнім оператором. Це означає, що після такого модуля, виділеного як цілий блок, можна поставити знак рівності для виведення значення функції. У блоці можуть міститися будь оператори та функції вхідної мови системи. Для передачі в блок значень змінних можна використовувати змінні документа, які поведуться в блоці як глобальні змінні.

Зазвичай модулю присвоюється ім'я зі списком змінних, після якого йде знак привласнення: $=$. Змінні в списку є локальними, і їм можна присвоювати значення при виклику функції, заданої модулем. Локальний характер таких змінних дозволяє присвоювати їм ті ж імена, що і глобальним змінним документа. Однак краще цього не робити і використовувати різні імена для локальних змінних програмних модулів і змінних документа.

На рис. 49 показано застосування операторів `onerror` і `return`, а також дія функції `errgr`, яка задає висновок запису в жовтому прямокутнику при активізації мишею виразу, що містить помилку.

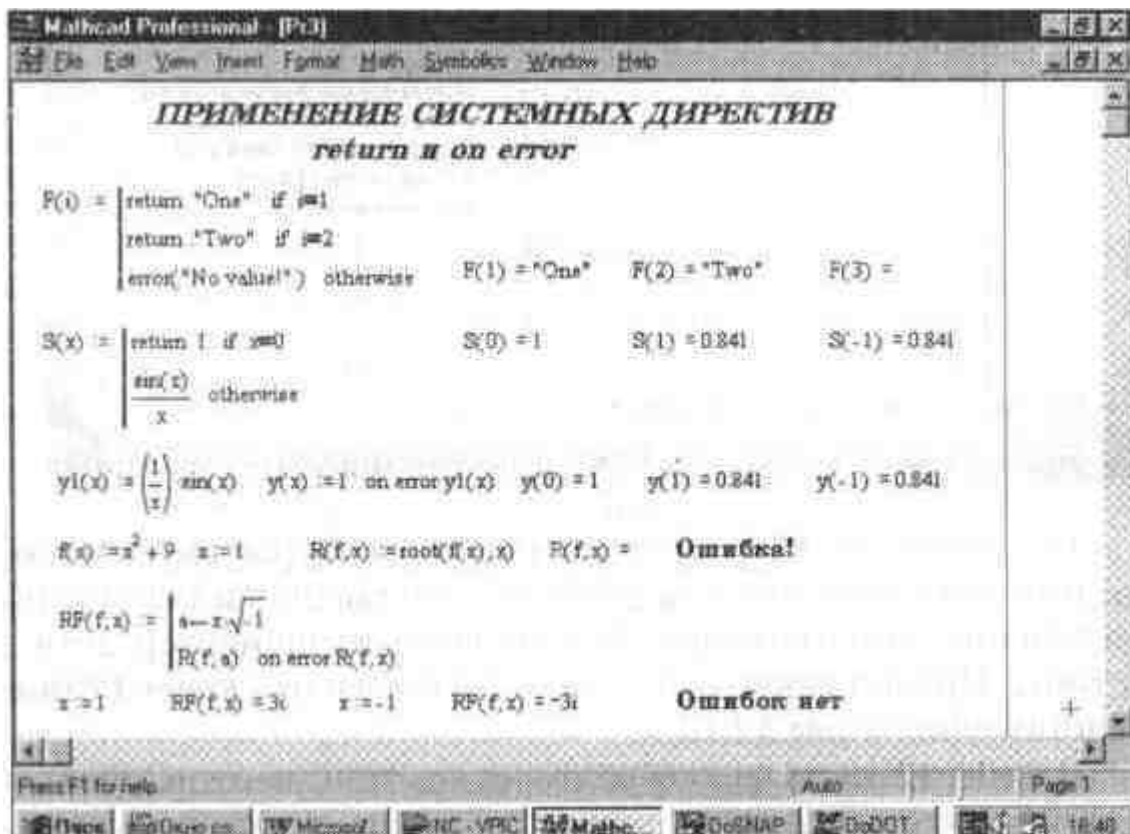


Рис. 49 - Застосування директив onerror і return в системі MathCAD

У цих прикладах треба звернути увагу на два моменти. Перший – можливість застосування оператора **onerror** поза програмного блоку для завдання функції $y(x) = \sin(x)/x$ з винятковим значенням 1 при $x = 0$. Другий – застосування цього оператора для вирішення рівняння, що має комплексні корені. Коли при такому рішенні задається дійсне початкове наближення для x , то функція $root(f, x)$ дає помилку. Вона виправляється за допомогою оператора **onerror** при завданні функції $RF(f, x)$, що має обробник помилок.

Завдання програмних модулів дозволяє реалізувати будь спеціальні прийоми програмування і може служити потужним засобом розширення системи шляхом завдання нових функцій.

Тепер розглянемо досить складний і навчальний приклад застосування програмного блоку. На рис. 50 показаний документ, в якому програмний модуль обчислює коефіцієнти Фур'є для заданої функції - пилообразного імпульсу (його можна міняти).

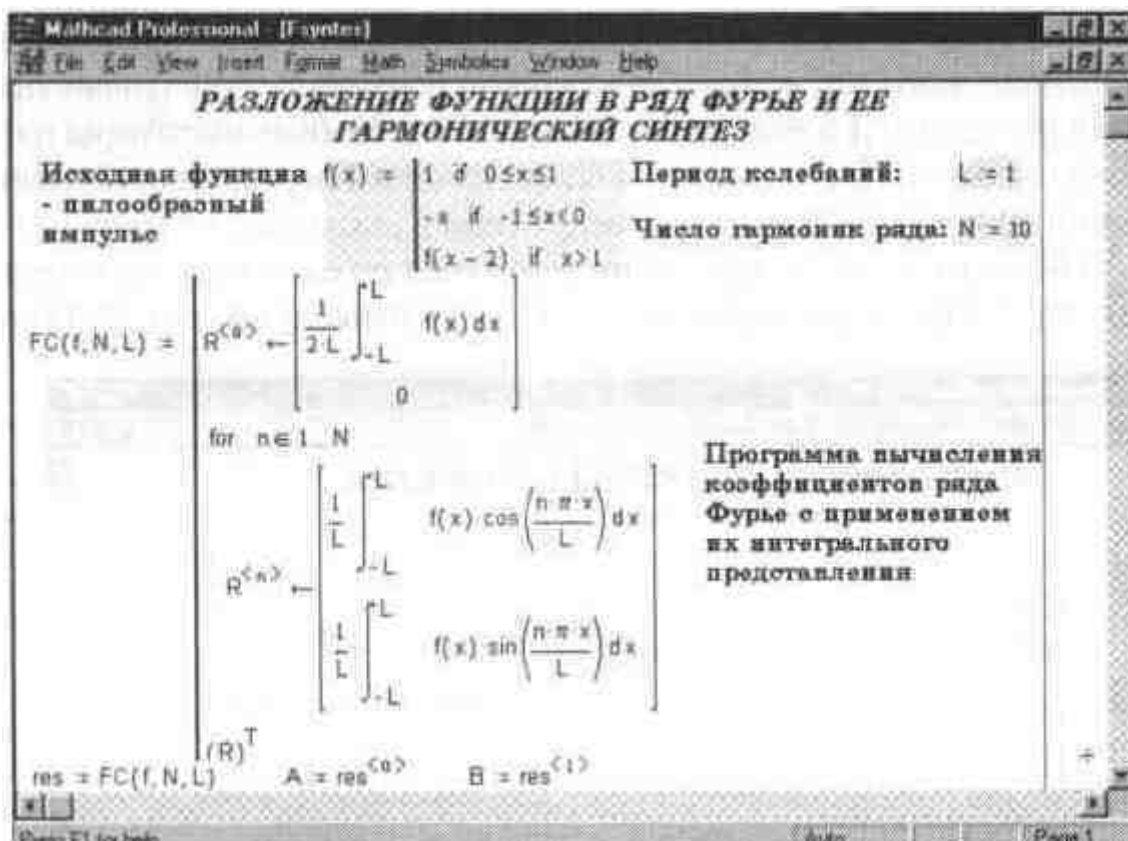


Рис. 50 - Обчислення коефіцієнтів A і B розкладання функції в ряд Фур'є в системі MathCAD

Тут обчислення коефіцієнтів ряду Фур'є виконано по їх класичним інтегральним уявленням. Це ілюструє можливість застосування в програмних блоках коштів вхідної мови системи. Продовження цього документа, що реалізує гармонійний синтез, представлено на рис. 51.

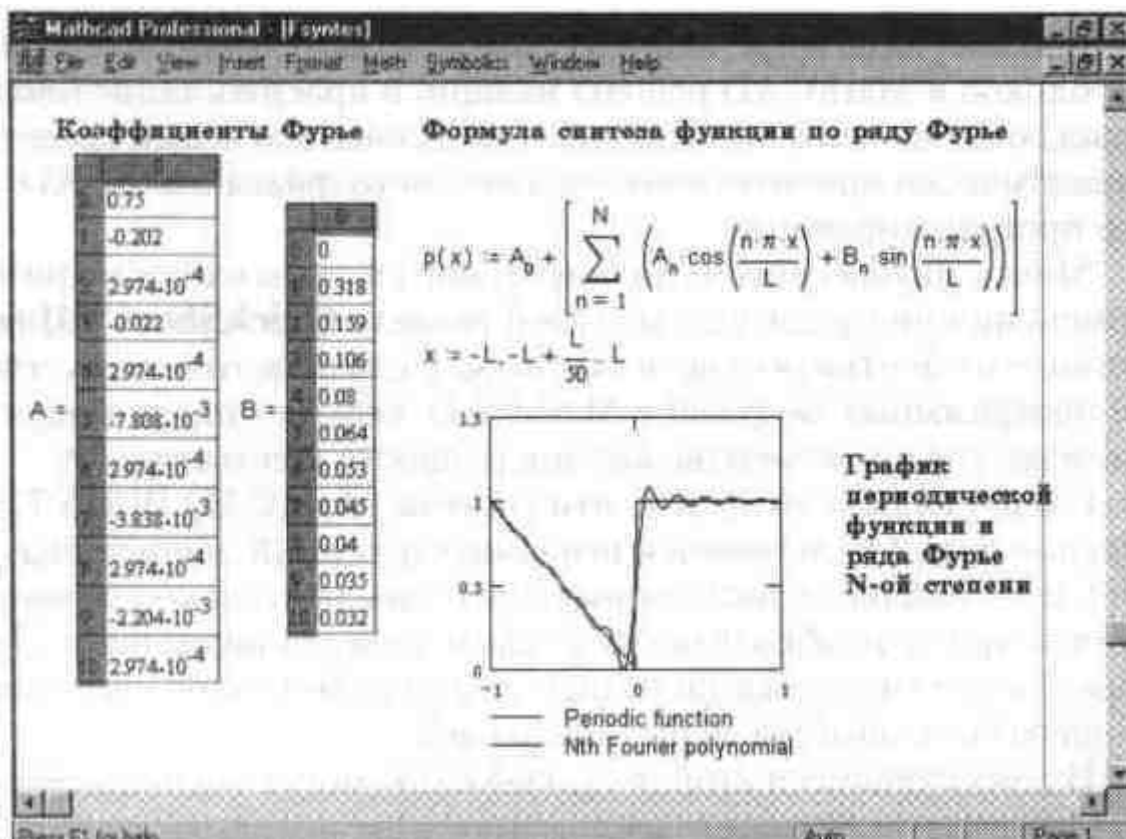


Рис. 51 - Реалізація гармонійного синтезу в системі MathCAD

На рисунку 51 показані вектори коефіцієнтів, наведена формула гармонійного синтезу вихідної функції по її гармонійному ряду і на графіку проведено зіставлення вихідної функції з її поданням поруч Фур'є з обмеженим числом гармонік N.

Ще один приклад застосування програмного модуля для завдання точок деякої функції в тривимірному просторі показаний на рис. 52. Там же представлено побудова цих точок всередині паралелепіпеда.

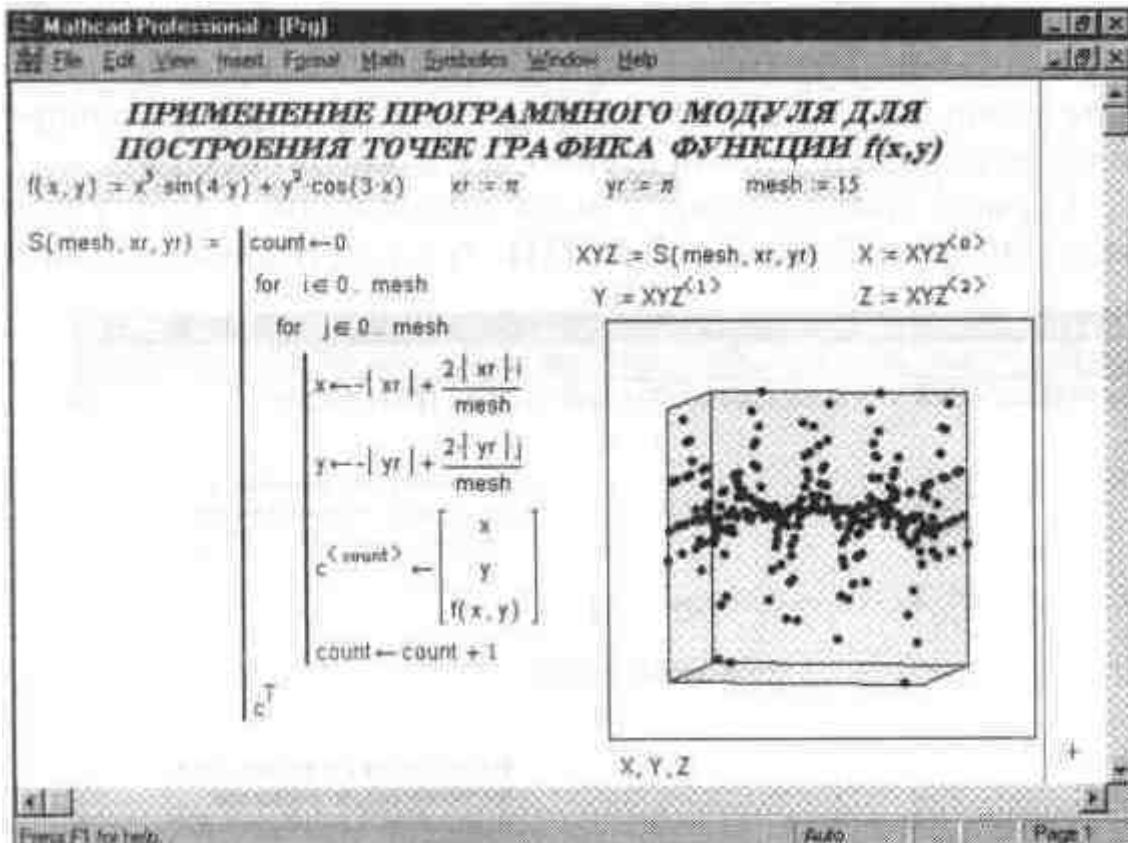


Рис. 52 - Застосування програмного модуля для побудови точок заданої функції в просторі в системі MathCAD

Зрозуміло, подібні завдання можуть в системі MathCAD вирішуватися і без використання в явному вигляді програмних засобів. Однак ці кошти полегшують вирішення складних завдань, особливо коли є опис їх програмної реалізації на якійсь мові програмування. Тоді нескладно перевести реалізацію рішення задачі з цієї мови на мову програмування системи MathCAD.

Взагалі, треба відзначити, що проблема включення в документи програмних блоків у MathCAD вирішена витончено і красиво, такі блоки часом просто прикрашають документи і дозволяють користуватися всіма засобами не тільки математично орієнтовано вхідної мови MathCAD, але й класичного програмування.

Література: [осн. 1, 4, дод. 2].

Завдання на СРС: Вивчення матеріалу лекції. Розробка завдань, які потребують програмування виконання обчислень. Приклади для розв'язання рівнянь з курсу вищої математики та фізики. **Література:** [осн. 1, 4, дод. 2].

Лекція 15

Тема: Основні принципи програмування в середовищі VBA. Створення макросів. Типи даних

VBA (*Visual Basic for Applications*) - це діалект мови Visual Basic, який розширює його можливості і призначений для роботи з додатками Microsoft Office і іншими додатками від Microsoft та інших фірм.

У більшість програм Microsoft Office вбудовано чудовий засіб, яке дозволить вам створювати програми, взагалі нічого не знаючи про програмування. Це засіб називається макрорекордером.

Макрорекордер, як зрозуміло з його назви, - це засіб для запису макросів. Макрос - всього лише ще одна назва для VBA-програми, а макрорекордер - засіб для його автоматичного створення.

Додатки Microsoft Office за замовчуванням налаштовані так, що не дозволяють запускати макроси. Тому перед тим, як приступати до створення макросів, в меню **Сервіс | Макрос | Безопасность** переставьте перемикач **«Уровень безопасности»** в положення **«Средняя»** або **«Низкая»**, а потім закрийте і знову відкрийте цю програму. Це потрібно зробити тільки один раз на початку роботи.

Принцип роботи макрорекордера найбільше схожий на принцип роботи магнітофона: ми натискаємо на кнопку – починається запис тих дій, які ми виконуємо. Ми натискаємо на другу кнопку – запис зупиняється, і ми можемо його програти.

Щоб створити макрос у макрорекордері VBA, необхідно:

1. У меню у стрічці оберіть вкладку **Вид** (або натиснути комбінацію клавіш **ALT+F8** та обрати у віконці, що з'явиться команду **«Создать»**, або у старих версіях Word 2003 і раніше оберіть вкладку **Сервіс | Макрос**), потім виберіть команду **«Макросы»** та натисніть на неї лівою клавішею миші, після чого з'явиться вниз випадаюче вікно вкладки **«Макросы»**, в якому оберіть **«Запись макроса»** (або у старих версіях Word 2003 і раніше оберіть **«Начать запись»**). З'явиться вікно **«Запись макроса»**, в якому вам буде потрібно визначити:

- *Ім'я макросу*. Правило таке: ім'я не повинно починатися з цифри, не повинне містити пробіли та символи пунктуації.

- *де зберегти макрос* (зазвичай –це *поточна книга*, або є можливість створити макрос одночасно зі створенням нової книги і особиста книга макросів PERSONAL.XLS, макроси з якої будуть доступні у всіх книгах).

2. Після натискання кнопки ОК або призначення кнопки або клавіатурної комбінації почнеться запис макросу. З'явиться маленька панель ■ **«Остановить запись»**.

Надалі у користувача може виникнути бажання внести в макрос будь-які доповнення або виправлення. Найпростіший, але незручний спосіб – скористатися у стрічці меню вкладки **Вид | Макросы | Изменить** або у старих версіях Word 2003 і раніше оберіть вкладку **Сервіс | Макрос | Макросы | Изменить**. Якщо ви користуєтеся макросом постійно, то можна використовувати найшвидший спосіб його виклику – клавіатурну комбінацію **<Alt> + <F11>**.

Повні можливості програмування в Office розкриваються при використанні редактора Visual Basic. Для входу в нього в меню **Сервіс | Макрос** вибрати **«Редактор Visual Basic»**, або натиснути клавіші **<Alt> + <F11>**.

У верхньому лівому кутку зазвичай відкрито вікно провідника проекту. Якщо воно випадково було закрито, то викликати його можна, натиснувши клавіші **<Ctrl> + <R>**.

Самий верхній рівень – це проект (Project), якому відповідає документ Word, робоча книга Excel і т. д.

Нижче розташовані програмні модулі. При роботі макрорекордера в Excel – Module1, куди і записуються всі створювані макрорекордером макроси.

Якщо вам потрібно створити програму вручну, а макросів в даному документі ще немає, то потрібно клацнути правою кнопкою миші по вузлу проекту (рядку, виділеній напівжирним шрифтом) і в контекстному меню вибрати команду **Insert | Module**. У проекті буде створено новий модуль і відкритий у вікні редактора коду. Починатися програма повинна з оголошення **Sub назва** ().

Редактор програмного коду – це, по суті, звичайний текстовий редактор, і в ньому ви можете вирізати і вставляти код, перетягувати або копіювати фрагменти коду. Але в нього вбудовано безліч доповнень, що допомагають набирати програму. Редактор сам контролює

синтаксичну правильність введених операторів і видає повідомлення про помилки, дописує за програміста ті ділянки програмного коду, які однозначно повинні фігурувати в даному контексті, автоматично розпізнає і виділяє різними кольорами синтаксичні конструкції VBA.

Основні синтаксичні принципи цієї мови наступні:

– VBA нечутливий до регістру (великими чи малими літерами набираєте текст у вікні VBA - однаково);

– щоб закоментувати код до кінця рядка, використовуються одинарні лапки (') або команда **REM**;

– для зручності читання можна об'єднати декілька фізичних рядків в один логічний за допомогою пробілу і знаку підкреслення після нього.

До основних конструкцій мови VBA відносяться *змінні, константи і службові слова (оператори, процедури, функції)*.

Змінні - це область оперативної пам'яті, призначена для зберігання будь-якого значення, яке може змінюватися в ході виконання програми. Перед роботою зі змінною рекомендується її оголосити. Приклад роботи з змінною:

```
Dim MyAge As Integer
```

```
MyAge = MyAge + 10
```

```
MsgBox MyAge
```

У VBA передбачено наступні **типи даних**:

- **Числові**:

- Integer - ціле число від -32 768 до 32 767;

- Long - велике ціле число від -2147483648 до 2147483647;

- Single і Double - значення з плаваючою комою (Double у 2 рази більше));

- **Рядкові** (String);

- **Логічний** (Boolean - може зберігати тільки значення True і False);

- **Об'єктний** (Object - зберігає посилання на будь-який об'єкт в пам'яті).

Можна оголосити кілька змінних в одному рядку, наприклад, так:

```
Dim nVar1 As Integer, s1 As String
```

Рядкові змінні записуються в подвійних лапках. Для злиття строкових змінних використовується оператор &.

Приклад:

```
Dim Name, Fam, St As String
```

```
Name = "Сергій"
```

```
Fam = "Петров"
```

```
St = "Розробник" & Fam & Name
```

Користувальницькі константи оголошуються за допомогою службового слова Const. Наприклад, Const Pi = 3.141592.

Література: [осн. 1, 3, дод. 1, 3].

Завдання на СРС: Вивчення матеріалу лекції. Процедури та макроси в VBA

Література: [осн. 1, 3, дод. 1, 3].

Лекція 16

Тема: Операції вводу-виводу в середовищі VBA. Математичні вирази.

Найпростіший спосіб прийняти інформацію від користувача – скористатися вбудованою функцією VBA Input Box ().

Повний її синтаксис виглядає так:

```
Input Box («Запрошення» [, «заголовок_вікна»] [, знач. за замовч.] )
```

Приклад:

Dim Name

Name = InputBox ("Введіть ім'я")

З'явиться віконце із запрошенням «Введіть ім'я» та кнопки OK і Cancel.

У більш повній формі, наприклад:

b1 = InputBox ("b1 =", "ширина каналу", 7)

віконце матиме заголовок "ширина каналу" і введене значення «7».

За замовчанням змінна, введена в віконце, має рядковий тип.

Range - дуже важлива функція. Повертає об'єкт **Range**, який являє собою діапазон комірок і використовується в Excel практично для будь-яких операцій з комірками.

Вибрати яку-небудь комірку, наприклад, A6: **Range ("A6"). Select** .

Або діапазон комірок:

Range ("A1: A6"). Select

або **Range (Cells (1, 1), Cells (6, 1)). Select**

Обраний діапазон може бути змінною типу **Range**, з яким можна надалі оперувати.

Наприклад посилання на діапазон комірок з A1 по D10 створюється так:

Dim R As Range

Set R = Worksheets ("Лист1").Range ("A1: D10")

Для вибору одної комірки можна використовувати функцію **Cells**:

x = **Cells** (6, 1).

або x = **Range ("A6").Value**

Самий простий спосіб вивести інформацію користувачеві - скористатися вбудованою функцією **VBA MsgBox** (Var).

Повний її синтаксис виглядає так:

MsgBox (Змінна [, кнопки] [, «заголовок_вікна»])

Приклад:

Result = MsgBox ("Натисніть кнопку", vbOKCancel, "Вікно повідомлення")

Опція Кнопки - наприклад, vbOKCancel видасть 2 кнопки: OK і Cancel. При натисненні OK значенням **Result** буде 1, при натисканні Cancel - 2.

Або просто: **Result = MsgBox** (g1, , "g1 =").

Або ще простіше: **MsgBox** (g1).

Вивести дані можна прямо в комірку.

Після того, як потрібна комірка знайдена, для запису в неї використовується властивість **Value**, наприклад:

x.**Value** = значення

Якщо комірка вже вибрана, можна записати: **ActiveCell.Value** = значення.

Або відразу вказати комірку виведення: **Cells** (6, 1) = значення.

Вибрати комірку, зміщену відносно поточної на і вниз та на j вправо:

ActiveCell.Offset (i, j).**Select**

Перш ніж щось виводити в таблицю, корисно очистити лист, для чого служить команда **Cells.Clear** .

Formula (або **FormulaR1C1**) = "="" дозволяє записати формулу в комірку.

Приклад:

Range("A3").Formula = "=2*\$A\$1^2+cos(\$B\$1)" (абсолютне посилання)

або "=2*A1^2+cos(B1)" (відносне посилання)

ActiveCell.FormulaR1C1 = "=2*RC[-1]^2+cos(RC[-1])"

У формулу можуть бути включені і змінні. Наприклад, замість 2 - використовуємо змінну a. Тоді попередню формулу слід переписати:

ActiveCell.FormulaR1C1 = "=" & a & "*RC[-1]^2+cos(RC[-1])".

В VBA потрібно знати синтаксис записів різноманітних функцій, які застосовуються в математичних виразах при обчислюваннях.

SQR () - квадратний корінь.

Наведемо деякі функції, які найчастіше застосовуються користувачами в VBA при обчислюваннях.

Функція **CStr ()** дозволяє перевести числове значення в строкове.

Функція **Val ()** – навпаки. При цьому функція читає дані зліва направо і зупиняється на першому нечисловій значенні. Це зручно, коли упереміж з числовими даними потрібно підписати одиниці виміру.

Abs () - повертає абсолютне значення числа;

Fix () - відкидає дробову частину числа;

sin (), cos (), tan () - відповідно синус, косинус, тангенс;

log () - натуральний логарифм.

Таблиця 5 - Приклади математичних виразів в VBA

Математичний вираз	Вираз, який записаний в VBA
$5x+12$	$5*x+12*y$
$\frac{x}{y}$	x/y
y^x	y^x
x	x
$19.55 \cdot 10^{-6}$	$19.55E-6$ або $19.55D-6$
$-a^b$	$-a^b$ або $-a^(b)$
a^{-b}	$a^(-b)$
a^{b+c}	$a^(b+c)$
$10^{-4.7}$	$10^(-4.7)$
$10^{4.7}$	$10^4.7$
$A \cdot B$	$A*B$
$A \cdot (-B)$	$A*(-B)$
a^{bc}	$a^(b*c)$
$(a^b)^c$	a^b^c або $(a^b)^c$
$\frac{a \cdot b}{c \cdot d}$	$a*b/(c*d)$ або $(a*b)/(c*d)$
$a \cdot 10^4$	$a*1E4$, $a*1D4$ або $a*10000$

Література: [осн. 1, 3, дод. 1, 3].

Завдання на СРС: Вивчення матеріалу лекції. **Література:** [осн. 1, 3, дод. 1, 3].
Конструктор вікон в VBA. **Література:** [осн. 1, 3, дод. 1, 3].

Лекція 17

Тема: Організація умовних операторів і циклів в середовищі VBA

Оператор **GoTo** - це оператор безумовного переходу, який застосовується коли хід виконання програми без перевірки будь-яких умов перестрибує на мітку в програмному коді.

Для мітки використовується натуральне число на початку рядка переходу або ім'я, яке ставиться перед рядком переходу і закінчується двокрапкою:

```
1 y = InputBox ("y")
2:
Cells (1, 1) = y
```

Найбільш поширений умовний оператор - **If**, який працює в парі з оператором **Then**.

Він може застосовуватись як в однорядковому, так і в блочному варіанті.

Однорядковий варіант застосовується, коли потрібно перевірити одну умову і в разі відповідності зробити якусь дію:

```
If Умова Then Команда 1 [Else Команда 2]
```

```
If y = 0 Then GoTo 1 Else GoTo 2
```

Якщо умови не виконані, програма виконує наступний рядок.

Повний синтаксис блоку **If**:

```
If Умова Then
```

```
Команди 1
```

```
.....
```

```
[ElseIf Умова N Then
```

```
Команди N]
```

```
[Else
```

```
Команди 2]
```

```
End If
```

Умова - вираз, що перевіряється на істинність. Якщо він істинний, то виконуються *Команди 1*, якщо хибний - *Команди 2*.

Оператори If ... Then можна вкладати один в одний:

```
If Var = 5 Then
```

```
MsgBox "Var = 5"
```

```
If Var = 10 Then
```

```
MsgBox "Var = 10"
```

```
End If
```

```
End If
```

Оператор **Select Case** застосовують для перевірки одного й того ж значення, яке потрібно багато разів порівняти з різними виразами:

```
Select Case Вираз
```

```
Case Умова 1
```

```
Команди 1
```

```
.....
```

```
[Case Умова N
```

```
Команди N]
```

```
[Case Else
```

```
Команди 2]
```

```
End Select
```

```
.....
```

Наприклад:

```
Select Case X
```

```
Case 0
```

```
y = -1
```

```
Case 1 To 10
```

```
y = X
```

Case 11 To 100, Is < 0

y = X^2

Case Else

y = SQR(X)

End Select

Існують засоби перехоплення та обробки помилок при виконанні операторів.

Загальний принцип обробки помилки виглядає так:

Перед небезпечним кодом (відкриття неіснуючого файлу, можливість ділення на нуль і т. п.) поміщається команда:

On Error GoTo мітка_обробника_помилки

Цикли використовуються в ситуаціях, коли нам потрібно виконати якусь дію кілька разів. Перша ситуація - ми знаємо, скільки разів потрібно виконати певну дію, в цьому випадку використовується конструкція **For ... Next**:

For змінна = поч.знач. To кін.знач. [Step крок]

.....

Next змінна

Якщо параметр **Step** не писати, то значення кроку буде дорівнювати одиниці.

Для циклу існують наступні обмеження:

- не можна всередині циклу перевизначати змінні циклу;
- не можна виходити з циклу назовні, поки він не закінчений;
- не можна заходити в цикл ззовні.

Для дострокового виходу з циклу при виконанні певних умов існує оператор **Exit**

For.

Цикли можуть бути вкладеними один в одне. Але вони не повинні перетинатися.

Коли ми не знаємо точно, скільки разів повинна бути виконана послідовність команд, використовуються конструкції *Do While умова ... Loop* і *Do Until умова ... Loop*.

У першому випадку цикл виконується, поки умова істинна, у другому - поки умова хибна.

Можна переписати цикл так, щоб умова перевірялося після завершення циклу:

Do Loop While умова або Loop Until умова

У цьому випадку цикл буде виконаний, принаймні, один раз.

Література: [осн. 1, 3, дод. 1, 3].

Лекція 18

Тема: Робота з масивами в середовищі VBA

Масив - це впорядкований набір даних, які зберігаються в послідовно розташованих комітках пам'яті і мають спільну назву. Масив складається з елементів, що мають назву масиву та індекс - порядковий номер.

За замовчуванням нижньою межею масиву є 0. Для установки відліку з 1 слід набрати команду **Option Base 1**.

Синтаксис оголошення масиву в середовищі VBA:

Dim ім'я (Розмір 1 [, розмір 2] [, розмір N]) As тип

Приклад:

Dim A(5) As Integer

A(2) = 20

MsgBox A(2)

Масиви можуть бути багатовимірними (до 60 вимірювань). Наприклад, двовимірний масив:

Dim A(5, 10)

В якості розміру може бути не тільки число, але і змінна або вираз. Наприклад:

n = 3

Dim A(n, n + 1)

Часто ми не знаємо наперед розміру масиву, і його можна змінювати в ході виконання програми. Такий масив називається динамічним. Його задають таким чином:

Dim A()

Встановити новий розмір і стерти попередні значення:

ReDim A(5)

Можна розмір і не вказувати, він буде підлаштовуватися під кількість введених значень.

Оператор **Array()** - дозволяє автоматично створити масив потрібного розміру і типу і одразу ввести в нього передані значення:

A = Array(10, 20, 30, 50, 70)

Для введення, виведення значень елементів масиву і роботи з ними використовують цикли. Для багатовимірних масивів застосовуються вкладені цикли з оператором **FOR**.

Приклад для введення елементів в двовимірний масив:

Dim A()

n = 3

m = 4

FOR i = 1 TO n

FOR j = 1 TO m

InputBox A(i, j)

NEXT j

NEXT i

Література: [осн. 1, 3, дод. 1, 3].

Лекція 19

Тема: Алгоритми. Види алгоритмів. Написання і графічне зображення алгоритмів

Алгоритм – це послідовна система команд, яка адресована виконавцю чітко і однозначно задавати процес вирішення завдання.

Одиницею вимірювання алгоритму є *крок*. Крок алгоритму є окремою замкінченою дією. Як правило алгоритм складається з декількох кроків.

Розрізняють такі *види алгоритмів*:

1. *Лінійні алгоритми* – складаються з декількох кроків, які виконуються послідовно один за одним.

2. *Розгалужений алгоритм* - має варіанти вибору умови, яка виконується. В такому алгоритмі виконання кроків змінюється від певних умов.




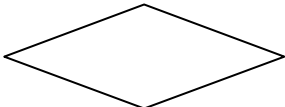

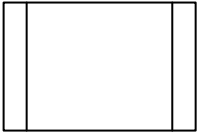
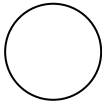
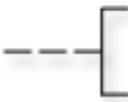
3. *Циклічні алгоритми* – це коли певна дія або група дій виконується декілька разів, циклічне повторення алгоритму декілька разів.

До етапу розроблення програм за допомогою любых мов програмування основним підготувальним етапом являється створення блок-схем за алгоритмом.

Блок-схема алгоритму – це графічне представлення логічної структури алгоритму, де кожний етап обробки інформації зображується у вигляді геометричних символів (блоків).

Графічне зображення блоків і їх призначення, що застосовуються у блок-схемах до програм за алгоритмом, представлено у табл. 6.

Таблиця 6 – Графічне зображення блоків у блок-схемах і їх призначення

Графічне зображення блоку	Призначення
	Початок та кінець алгоритму
	Блок вводу вхідних даних та виведення результату. Цей символ не визначає носія даних
	Виконання однієї або кількох операцій, обробка даних. Всередині фігури записують безпосередньо самі операції.
	Блок умови. В цьому блоці пишеться умова, в залежності від якої вибираються напрямки дії алгоритму. На лініях, що розгалужуються від блоку, пишеться „Так” або „Ні” – залежно від виконання записаної умови.
	Початок і кінець циклу. Межі змінної циклу і крок її збільшення записуються всередині символу початку, а в символі кінця циклу – змінна циклу. Операції, що виконуються всередині циклу, розміщуються між ними.
	Виконання процесу, що складається з операцій, які визначені в іншому місці програми (у підпрограмі, модулі). Всередині символу записується назва процесу і дані, які передані в нього.
	З'єднання. Відображає вихід в частину схеми і вхід з іншої частини цієї схеми. Використовується для обриву лінії та продовження її в іншому місці. Всередині ставиться позначка переходу.
	Коментар. Використовується для детальнішої інформації про процес, коли текст в символі перевищує його обсяг.

Існують правила зображення блок-схем алгоритмів. Кожен алгоритм має початок та кінець. Кожна команда алгоритму представляється у вигляді геометричних символів,

які мають певну конфігурацію, в залежності від характеру дій, що будуть виконуватись. Геометричні символи з'єднуються між собою лініями або лініями зі стрілками, які вказують порядок виконання дій у блок-схемі алгоритму.

Приклад: Фрагмент блок схеми для розгалуженого алгоритму представлений на рисунку 55, якщо задана умова для функції.

$$y = \begin{cases} x, & \text{якщо } x \geq 0 \\ -x, & \text{якщо } x < 0 \end{cases}$$

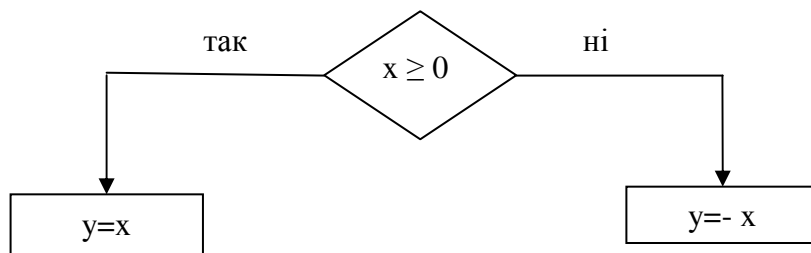


Рис. 55 – Фрагмент блок-схеми розгалуженого алгоритму

Приклад: Блок схема для лінійного алгоритму, якщо задано знайти суму S з трьох доданків a, b, c (Рис. 56).

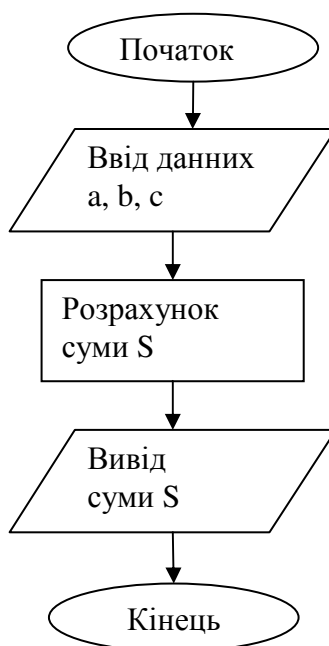


Рис. 56 – Блок схема для лінійного алгоритму знаходження суми S

Література: [осн. 1, 3, дод. 1, 3].

Завдання на СРС: Вивчення матеріалу лекції. Розробка лінійних алгоритмів. Розробка розгалужених і циклічних алгоритмів та їх запис у вигляді блок-схем.

Література: [осн. 1, 3, дод. 1, 3].

Лекція 20

Тема: Робота в електронному Кампусі НТУУ «КПІ ім. Ігоря Сікорського»

Електронний кампус НТУУ «КПІ ім. Ігоря Сікорського» входить складовою в Корпоративний портал НТУУ «КПІ ім. Ігоря Сікорського», що є підсистемою Єдиного Інформаційного Середовища НТУУ «КПІ ім. Ігоря Сікорського».

Нижче подано скорочення, які часто вживаються в цій лекції:

ЕК – електронний кампус

ЄІС НТУУ «КПІ ім. Ігоря Сікорського» - Єдине інформаційне середовища НТУУ «КПІ»

ПВК – персональний віртуальний кабінет

Призначення системи ЕК НТУУ «КПІ ім. Ігоря Сікорського»:

- Інформатизація навчального процесу університету з метою підвищення якості навчання студентів.

- Забезпечення навчального процесу сучасними інформаційними технологіями.

Основа для розробки системи ЕК НТУУ «КПІ ім. Ігоря Сікорського»:

- Наказ по НТУУ «КПІ» № 4-238 від 03.08.2009 р. «Про створення функціонального Інтернет-порталу Електронних Інформаційних Ресурсів університету».

- Наказ по НТУУ «КПІ» № 4-239 від 09.08.2009 р. . «Про створення Банку веб-ресурсів навчальних дисциплін НТУУ «КПІ» як складової Єдиного інформаційного середовища НТУУ «КПІ»».

Головні завдання, що вирішуються системою ЕК НТУУ «КПІ ім. Ігоря Сікорського»:

- Інформаційна підтримка навчального процесу кафедр університету.

- Забезпечення електронного спілкування між учасниками навчального процесу.

Для вирішення поставлених завдань в системі ЕК НТУУ «КПІ ім. Ігоря Сікорського» розроблені *віртуальні кабінети за профілями користувачів*: студент (додаткові функції профілю: староста, профорг), викладач-науковець (додаткові функції профілю: куратор), методист кафедри, завідувач кафедрою.

Вхід до персонального віртуального кабінету системи ЕК НТУУ «КПІ ім. Ігоря Сікорського»

Роботу з ЕК НТУУ «КПІ» підтримують всі поширені браузері (програмне забезпечення для перегляду веб-сайтів), такі як Internet Explorer, Mozilla Firefox, Opera і Google Chrome.

Подальші інструкції по роботі з ЕК НТУУ «КПІ» наводяться на прикладі браузера Mozilla Firefox 7.0.1.

Щоб увійти до персонального віртуального кабінету (ПВК) системи ЕК НТУУ «КПІ», необхідно:

- Натиснути кнопку Пуск, перейти до пункту Програми (або Усі програми), вибрати програму (браузер) Mozilla Firefox;

- В адресному полі браузера Mozilla Firefox ввести URL-адресу сторінки входу в систему: **http://login.kpi.ua** (Рис. 57), і натиснути клавішу Enter на клавіатурі;

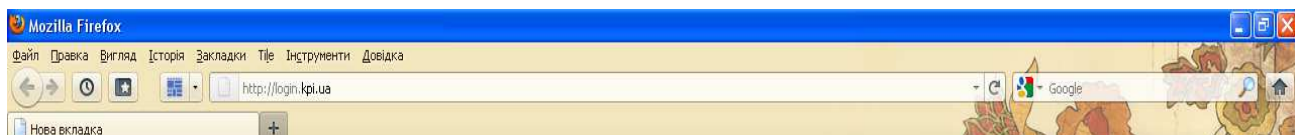


Рис. 57 - Ввід URL-адреси в адресне поле браузера

- На сторінці, що завантажилась (Рис.58), необхідно натиснути посилання «Вхід» (верхній правий кут сторінки), для авторизації в системі ЕК НТУУ «КПІ»;



Рис. 58 - Сторінка системи ЕК НТУУ «КПІ»

- У вікні, що з'явилося (Рис.59), вводяться логін та пароль, які були видані відповідальним за впровадження системи ЕК НТУУ «КПІ» на підрозділі, після чого натискається кнопка «Увійти»;

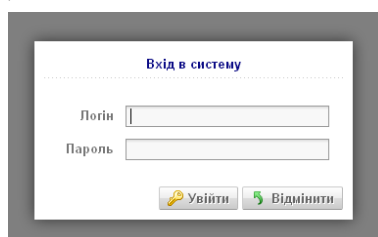


Рис. 59 - Вікно авторизації користувача в системі ЕК НТУУ «КПІ»

- Після успішної авторизації (Рис.60) для входу до ПВК системи ЕК НТУУ «КПІ» необхідно натиснути кнопку «Електронний кампус НТУУ «КПІ»» (з лівої сторони вікна).

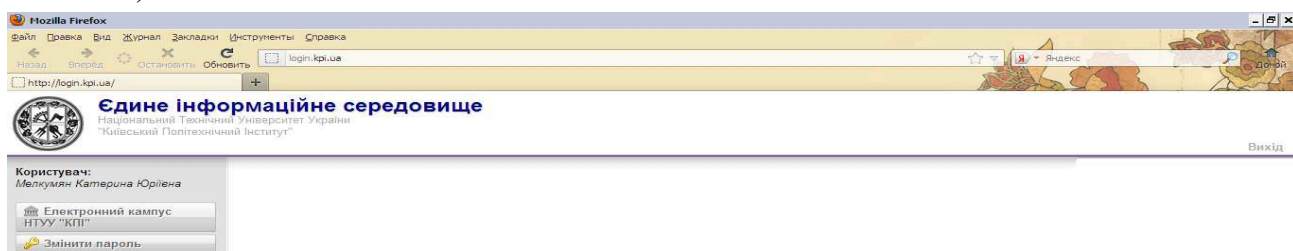


Рис. 60 - Сторінка авторизованого користувача системи ЕК НТУУ «КПІ»

Після чого здійснюється завантаження ПВК (Рис. 61).

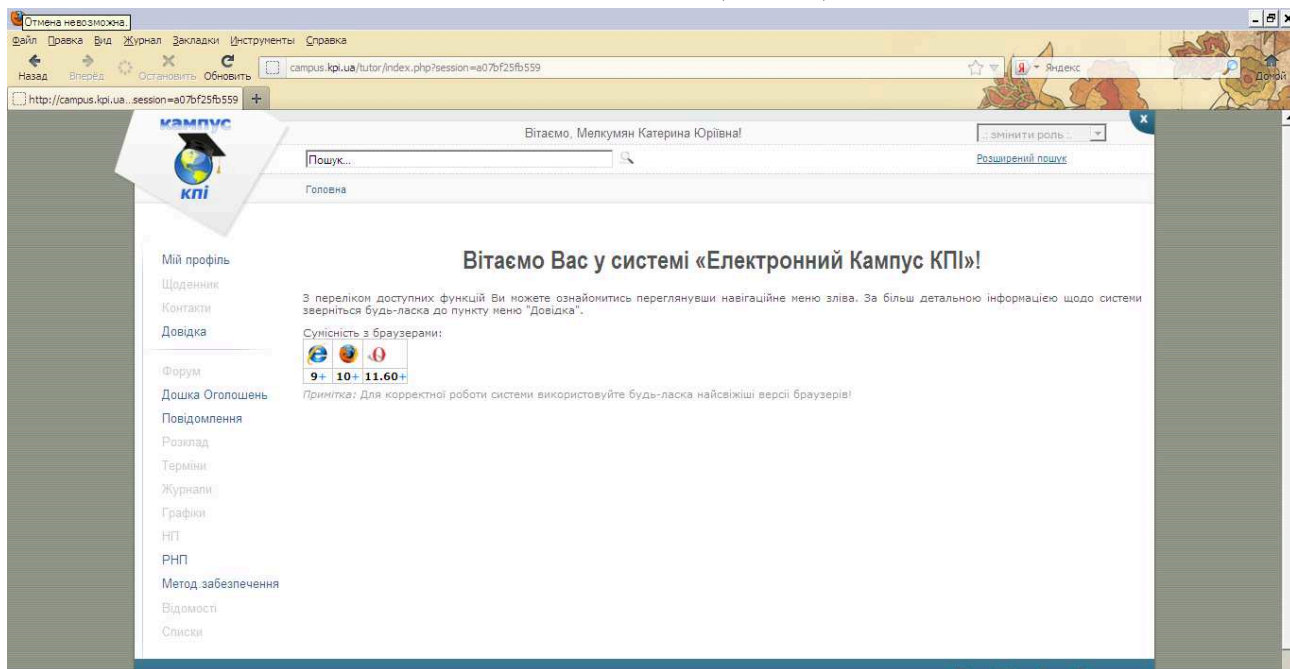



Рис. 61 - Головна сторінка ПВК користувача в системі ЕК НТУУ «КПІ»

На головній сторінці присутні посилання на завантаження останніх версій браузерів, які забезпечують коректну роботу системи.

Вихід із системи ЕК НТУУ «КПІ ім. Ігоря Сікорського»

Для повернення до сторінки авторизованого користувача (Рис. 60) необхідно натиснути кнопку вихід , що розташована у верхній правій частині головної сторінки ПВК (Рис. 61) системи ЕК НТУУ «КПІ».

Для виходу з системи ЕК НТУУ «КПІ» необхідно натиснути посилання **Вихід**, що розташовано у верхній правій частині сторінки авторизованого користувача (Рис. 60).

Розділи персонального віртуального кабінету системи ЕК НТУУ «КПІ ім. Ігоря Сікорського»

ПВК має уніфіковану структуру (Рис. 61), саме тому не залежить від профілю користувача.

На головній сторінці ПВК системи ЕК НТУУ «КПІ» відображаються основні розділи: мій профіль, щоденник, контакти, довідка, форум, дошка оголошення, повідомлення, розклад тощо.

Робота у розділі «Повідомлення»

В даному розділі надається можливість обміну повідомленнями між користувачами системи «Кампус».

Повідомлення – короткий текст, який відправляємо адресату/адресатам.

Адресат – співробітник, або студент, що отримує повідомлення.

Щоб розпочати роботу з повідомленнями, необхідно на Головній сторінці натиснути на розділ «Повідомлення» (Рис.62).

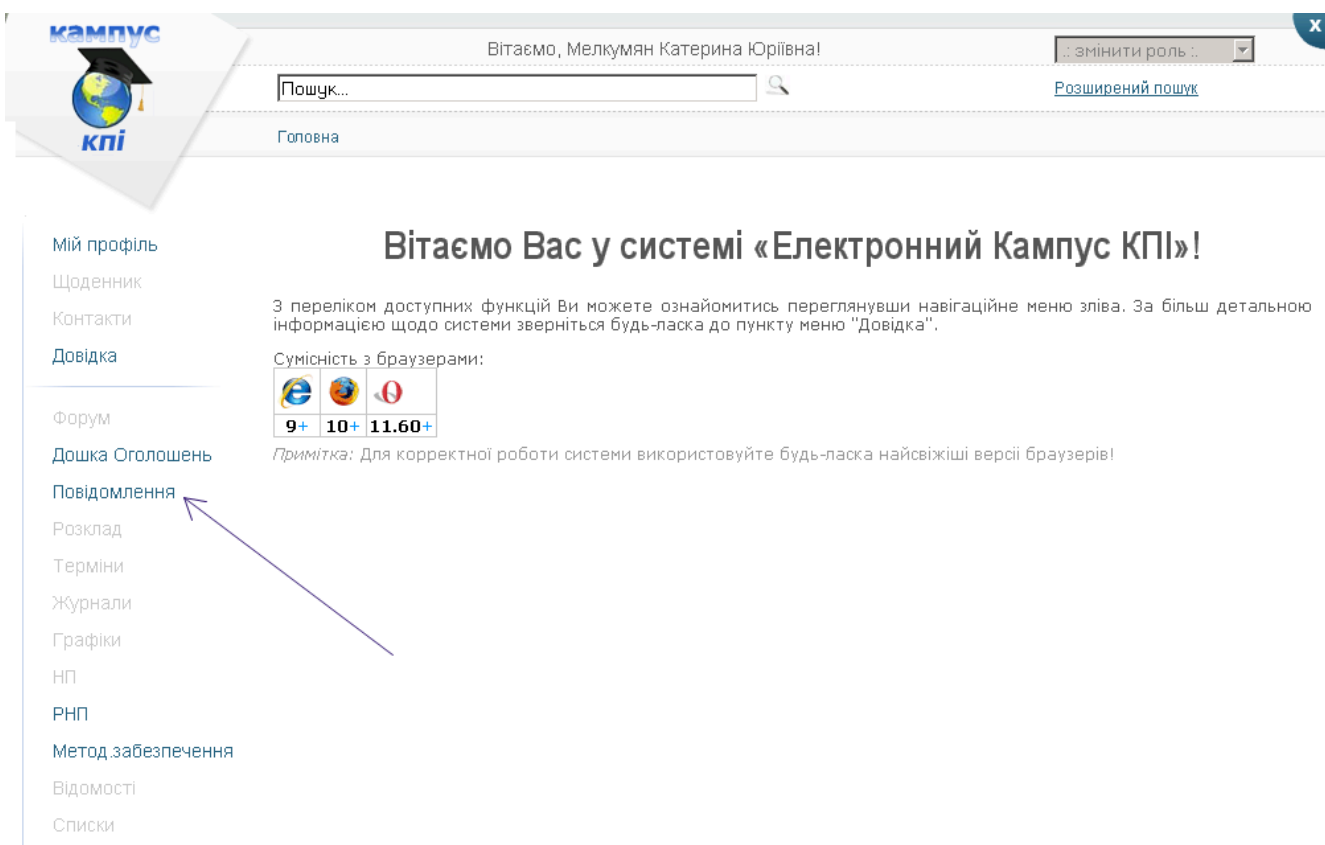


Рис. 62 – Виклик функції «Повідомлення»

Сторінка, що завантажилась, - є головною сторінкою «Повідомлень» (Рис.63).

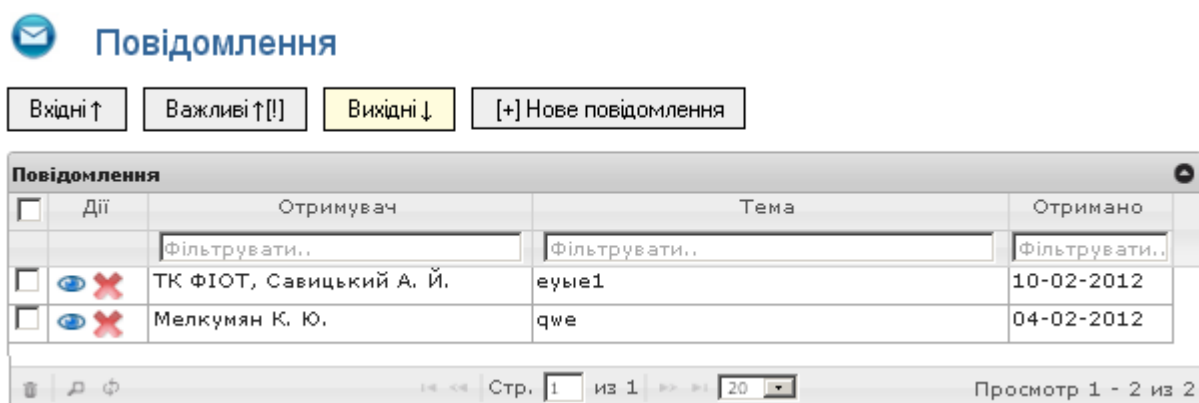


Рис. 63 - Головна сторінка «Повідомлень»

Користувач в своєму ПК має можливість робити такі дії з повідомленнями:

- Переглянути повідомлення.
- Відповісти на отримане повідомлення.
- Видалити отримані повідомлення.
- Створити нове повідомлення.

Розглянемо дії з повідомленнями детальніше.

Перегляд повідомлень

Натискаючи пункти меню «Вхідні», «Важливі», «Вихідні» у верхній панелі, викликаємо відображення відповідних груп повідомлень.

Якщо кількість повідомлень перевищує розміри вікна – слід скористатися «прокруткою» у правій його частині. Також можна користуватися клавішами управління у нижній частині вікна (Рис.64).



Рис. 64 – Кнопки управління повідомленнями

Для зручності можемо регулювати кількість повідомлень, що відображаються на одній сторінці.

Користувач має можливість звернути і згодом розвернути вікно повідомлень, натиснувши відповідну кнопку.

Для перегляду повідомлення необхідно:

- клікнути на тему листа;
- натиснути на «око» зліва від повідомлення (Рис.65).

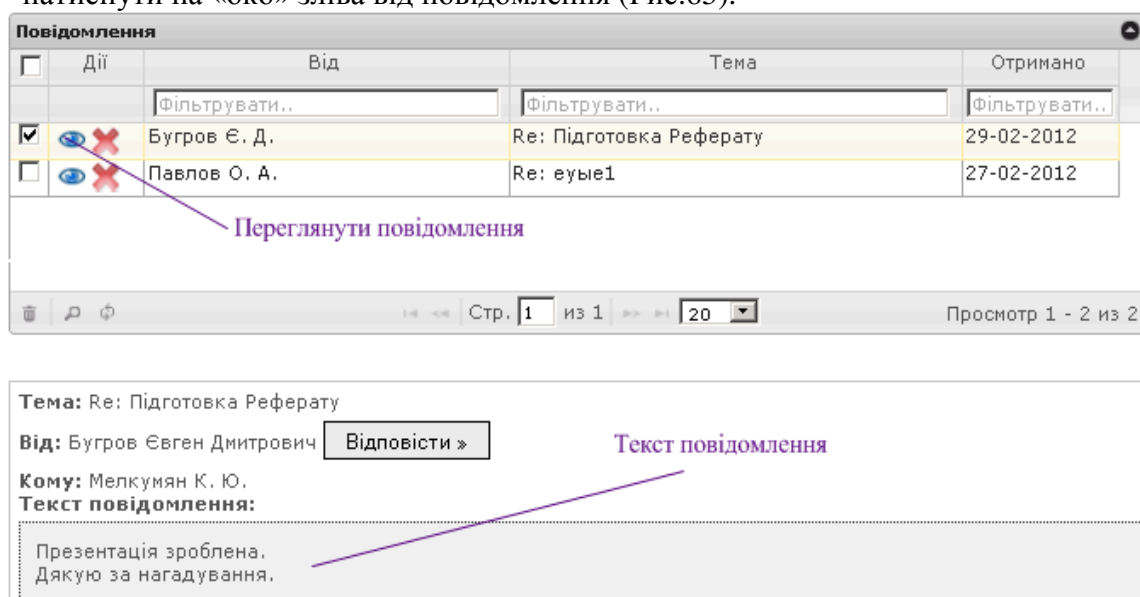


Рис. 65 – Перегляд повідомлення

Відповідь на повідомлення

Переглянувши «вхідне», або «важливе» повідомлення, маємо можливість відповісти на нього.

Для цього, у вікні перегляду повідомлення, слід натиснути кнопку «Відповісти» (Рис.65). Відкриється вікно (Рис.66), в якому необхідно заповнити поля і натиснути «Відіслати». Якщо не всі дані внесені, при спробі відправлення, з'явиться повідомлення (Рис.67). Треба внести дані яких не вистачає і знову спробувати «Відіслати».



Повідомлення

Нове повідомлення
(підтримується масове відсилання повідомлень подібно е-мэй скринькам)

Одержувач(і) *

Бугров Євген Дмитрович

Пріоритет

Низький

Тема повідомлення *

Re: Re: Підготовка Реферату

Текст повідомлення *

Заповнити послідовно

«Відмовитись» «Відіслати»

Коли внесена необхідна інформація натиснути "Відіслати"

Повернутись до попереднього вікна


Рис. 66 – Відповідь на отримане повідомлення

Помилка! Перевірте будь-ласка коректність заповнення полів повідомлення!

Рис. 67 – Повідомлення при некоректному внесенні даних

Видалення повідомлення

В ПВК можна видалити як одне так і сразу декілька повідомлень.





Для видалення окремого повідомлення необхідно натиснути  (Рис.68).






Повідомлення

Вхідні ↑ Важливі ↑(!) Вихідні ↓ [+] Нове повідомлення

Повідомлення

<input type="checkbox"/>	Дії	Від	Тема	Отримано
		Фільтрувати..	Фільтрувати..	Фільтрувати..
<input type="checkbox"/>	 	Бугров Є. Д.	Re: Підготовка Реферату	29-02-2012
<input type="checkbox"/>	 	Павлов О. А.	Re: euyel	27-02-2012

Видалити

Стр. 1 из 1 20

Просмотр 1 - 2 из 2

Рис. 68 – Видалення окремого повідомлення

Для видалення декількох повідомлень одночасно необхідно виділити їх і натиснути «корзину» (Рис.69).

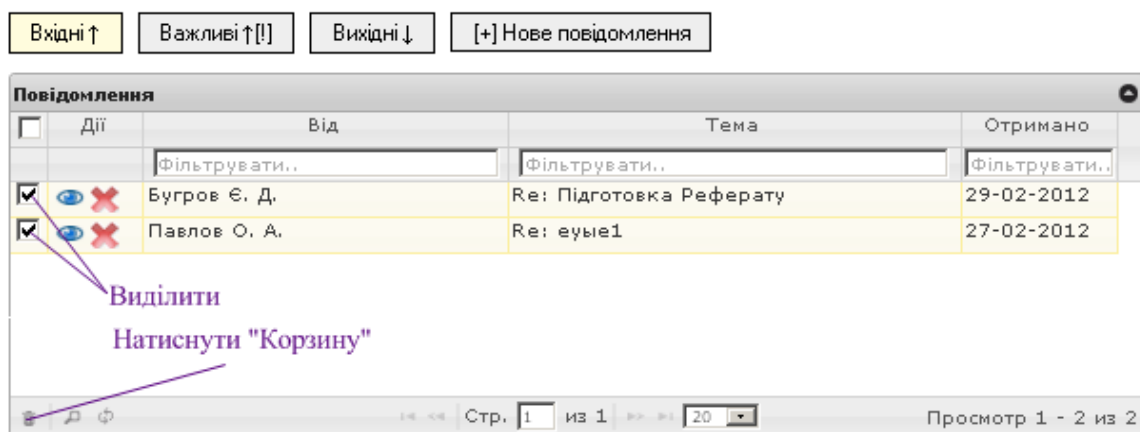


Рис. 69 – Видалення кількох повідомлень

Коли натискаємо «видалити» («хрестик», або «корзину») -- викликається вікно підтвердження видалення (Рис.70).

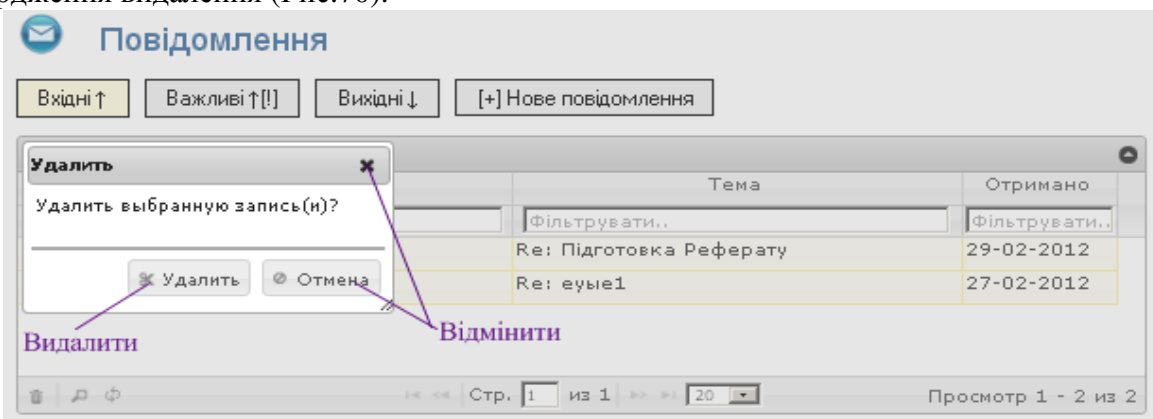


Рис. 70 – Підтвердження видалення повідомлення

Створення повідомлення

Користувача в своєму ПВК має можливість створити:

- Індивідуальне повідомлення: одній, або декільком особам.
- Повідомлення для групи осіб: студентській групі/групам, співробітникам кафедри, факультету.

Для **створення повідомлення** слід натиснути кнопку «Нове повідомлення» на головній сторінці «Повідомлення» (Рис.63).

Для **відміни створення** «Нового повідомлення» і повернення до попереднього вікна – натиснути «Відмовитись» (Рис.71).

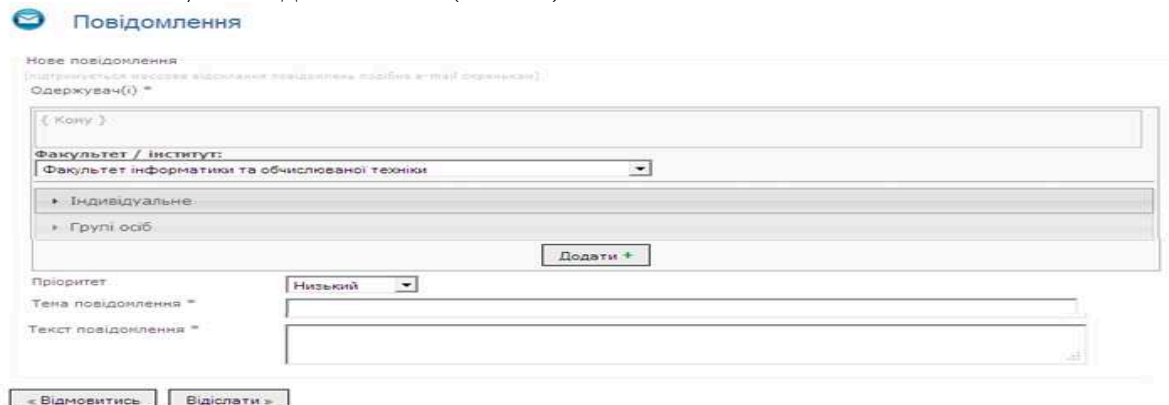


Рис. 71 - Відміна створення «Нового повідомлення»

«Індивідуальне» повідомлення

Для створення натискаємо відповідну кнопку («Індивідуальне») (Рис.72).



Рис. 72 – Пункти меню «Нового повідомлення»

Завантажується форма створення нового індивідуального повідомлення (Рис.73).

На даній формі необхідно :

- Вибрати одержувача(ів) повідомлення:
 - факультет/інститут особи, якій бажаємо відправити повідомлення;
 - статус особи: співробітник/студент (розглянемо спочатку випадок, коли особа – співробітник);
 - кафедра, на якій працює особа;
 - прізвище, ім'я, по-батькові;
 - натиснути кнопку «Додати» при створенні повідомлення для декількох адресатів. Для видалення зайвого адресата – натиснути «хрестик».
- Написати тему, текст повідомлення.
- Натиснути кнопку «Відіслати».

Рис. 73 – Форма створення нового індивідуального повідомлення

Якщо у створенні повідомлення була допущена помилка – після натиснення кнопки «Відіслати» -- з'явиться відповідний текст (Рис.74).



Повідомлення

Помилка! Перевірте будь-ласка коректність заповнення полів повідомлення!

Рис. 74 - Помилка при відсилання повідомлення

Групове повідомлення

Для створення натискаємо відповідну кнопку («Групі осіб») (Рис.72).

Завантажується форма створення нового групового повідомлення (Рис.75). На даній формі необхідно :

- Вибрати групу одержувачів повідомлення:
 - кафедру факультету/інституту, співробітникам якої бажаємо відправити повідомлення;
 - навчальну(і) групу(и) кафедри факультету/інституту, студентам якої бажаємо відправити повідомлення;
 - натиснути кнопку «Додати» при створенні повідомлення для декількох груп адресатів. Для видалення зайвого адресата – натиснути «хрестик».
- Написати тему, текст повідомлення.
- Натиснути кнопку «Відіслати».

Рис.75 – Форма створення нового групового повідомлення

Робота у розділі «Робочий навчальний план» (РНП)

Розділ РНП («Робочий навчальний план») системи «Кампус» дозволяє користувачу переглянути навчальні плани відповідно його потреб та повноважень.

Для використання відповідних функцій слід відкрити вкладку (Рис.76):



З'явиться головне вікно розділу.

Обираємо загальні дані щодо інформації яку бажаємо переглянути

Обираємо тип інформації

Згорнути/розгорнути вікно

Натиснувши посилання переглянути можемо Методичне забезпечення щодо відповідного модулю

Згорнути/розгорнути

Прокрутка

№ п/п	Найменування кредитного модулю	Назва кафедри	Обсяг дисциплін		Аудиторні години					Самостійна робота студентів
			Кредитів ECTS	Годин	Всього	Лекції	Практичні (семинарські)	Лабораторні (комп. практикум)		
1	Іноземна мова	англійської мови гуманітарного спрямування №2 ФЛ	1	36	4	...	4	...	32	
3	Іноземна мова професійного спрямування	англійської мови гуманітарного спрямування №2 ФЛ	1.5	54	4	...	4	...	50	
18	Інформаційні природно-мовні технології	технічної кібернетики ФІОТ	4	144	14	8	...	6	130	

Рис. 76 – Система «Електронний Кампус КПІ ім. Ігоря Сікорського»

Далі необхідно ввести загальні дані щодо інформації, яку бажаємо отримати (навчальний рік, випускна кафедра, освітньо-кваліфікаційний рівень і т. д.) → Обрати тип інформації («по замовчуванню» відображаються аудиторні години).

Тепер, за необхідності, можливість маємо обрати студентську групу, якщо тільки щодо неї інформація нас зараз цікавить. Повернутись до більш повного показу можливо знову натиснувши необхідний «тип інформації» (наприклад, «Аудиторні години»).

Всі вимоги, що ми відображаємо у налаштуваннях відображення даних, негайно викликають необхідний план.

Якщо певний кредитний модуль у вікні має вигляд посилання, значить, натиснувши на нього, можемо отримати необхідне методичне забезпечення (статті, методички і т. д.).

Робота у розділі «Методичне забезпечення (МЗ)»

Даний модуль призначено для надання користувачеві можливості розмішувати свої інформаційні ресурси (статті, «методички» і т. д.) у системі «Кампус», а також завантажувати матеріали інших користувачів, відповідно повноважень.

Методичне забезпечення (МЗ) – частина електронних інформаційних ресурсів (ЕІР), що відноситься до забезпечення навчального процесу НТУУ «КПІ».

ЕІР в електронному кампусі (ЕК) складається з:

- *картки ЕІР* (представляє собою опис ЕІР в єдиному сховищі);

- *приєднаних ЕІР* (представляє собою набір приєднаних електронних ресурсів. Електронні ресурси можуть бути представлені файлами (форматів *.doc, *.docx, *.xls, *.pdf, *.ppt і т.і.) та посиланнями на ресурс в інтернеті, на серверах кафедр.

Робота з МЗ складається з двох етапів:

Етап 1. Реєстрація ЕІР в системі ЕК – мається на увазі заповнення картки ЕІР (опису ЕІР).

Етап 2. Приєднання ЕІР до картки ЕІР.

Картка ЕІР складається з наступних полів (Рис.77):

1) Рівень доступу до ЕІР – визначає розділ ЕК в якому буде відображено ЕІР на сторінках ЕК. Визначення відповідного розділу впливає на видимість ЕІР для користувачів ЕК. Виділяються наступні розділи:

[Кредитного модулю](#) | [Кафедральні](#) | [Інституту/Факультету](#) | [Загальноуніверситетські](#)

Кредитного модулю – МЗ, до якого мають доступ лише викладачі, на кафедрах у яких читається

Кафедральні –

Інституту/факультету –

Загальноуніверситетські -

2) Вид ЕІР – обов’язкове для заповнення поле, яке дозволяє класифікувати ЕІР в ЕК.

3) Блок «Загальні відомості» містить поля загального опису: назва, яку побачать користувачі на сторінках кампусу, короткий пояснюючий опис ЕІР (за бажанням), дата створення (УВАГА! Це не є датою публікації ЕІР в ЕК, хоча може збігатися з нею. Це є датою саме створення ресурсу).

Для доступу необхідно натиснути (Рис.78):

Мій профіль

Щоденник

Контакти

Довідка

Форум

Дошка Оголошень

Повідомлення

Розклад

Терміни

Журнали

Графіки

НП

РНП

Метод.забезпечення

Вітаємо Вас у системі «Електронний Кампус КПІ»!

З переліком доступних функцій Ви можете ознайомитись переглянувши навігаційне меню зліва. За більш детальною інформацією щодо системи зверніться будь-ласка до пункту меню "Довідка".

Сумісність з браузерами:



Примітка: Для коректної роботи системи використовуйте будь-ласка найсвіжіші версії браузерів!

Рис.77 - Картка ЕІР складається з наступних полів

Рис. 78 - Методичне забезпечення

Щоб переглянути певний ресурс, слід спочатку вибрати його рівень (Рис.79):

Рис. 79 - Методичне забезпечення та його рівні

Далі необхідно обрати підрозділ до якого відноситься необхідний ресурс
(Рис.80):

Рис.80 - Методи забезпечення та його підрозділи

Щоб повернутися до попереднього вікна – натиснути «Метод. забезпечення» на лівій панелі або скористатися рядком навігації, що відображає історію переходів системи (Рис.81).

Рис. 81 - Повернутися до попереднього вікна

Література: [осн. 5].

Лекція 21

Тема: Створення презентацій в MS PowerPoint

Презентації, підготовлені з допомогою програми MS PowerPoint, стають все більш популярним засобом представлення інформації, створення конспектів лекцій та супроводу публічних виступів. Для того щоб така презентація дійсно робила виступ більш ефективним, необхідне дотримання певних вимог. Ці вимоги здаються очевидними; тим не менш, практика показує, що вони часто порушуються не тільки початківцями, але і кваліфікованими користувачами і фахівцями в області інформаційних технологій.

1. Інформація на слайдах має добре читатися

1.1. Колірне рішення презентації.

Незалежно від того, наскільки барвисті і привабливі шаблони оформлення слайдів, пропонувані програмою MS PowerPoint, вибір колірного рішення — співвідношення кольорів фону слайда і текст — диктується умовами показу. Для демонстрації презентації на екрані монітора або за допомогою проектора в добре затемненому приміщенні цілком виправданий вибір яскравих кольорів, темного фону слайдів і світлого кольору тексту. Типовою ситуацією є відсутність затемнення, тому оптимальним для електронної презентації є світлий фон слайдів і темний колір тексту.

1.2. Розмір і тип шрифту.

Розмір шрифту, дозволяє зробити текст зручним для читання на екрані, передбачений шаблонів презентацій, тому має сенс розміщувати на слайді такий обсяг тексту, який би не приводив до автоматичного зменшення розміру шрифту. Крім того, потрібно пам'ятати, що на екрані краще сприймаються шрифти без зарубок (такі як, наприклад, Tahoma, Verdana, Arial), тому використовувати звичний для друкованих текстів шрифт Times New Roman у презентаціях не рекомендується, також не рекомендується використовувати курсивне накреслення.

1.3. Обсяг і структурування інформації на слайді.

Текст презентації не повинен служити конспектом для доповідача — для цього в програмі MS PowerPoint існує спеціальний режим підготовки приміток до слайдів. Розгорнуті тексти на слайдах презентації — конспектів лекцій цілком виправдані, але в презентаціях, що ілюструють публічний виступ, текст повинен бути згорнутий до ключових слів і фраз. Повні розгорнуті пропозиції на слайдах таких презентацій використовуються тільки при необхідності цитування.

Списки на слайдах не повинні включати більше 5-7 елементів. Якщо елементів списку все-таки більше, їх краще розташувати в дві колонки.

В таблиці не повинно бути більше 4 рядків і 4 стовпців — в іншому випадку дані в таблиці буде просто неможливо побачити. Клітинки з назвами рядків і стовпців і найбільш значущі дані рекомендується виділяти кольором.

Гістограми не повинні включати більше 4 категорій, а організаційні діаграми — більше 5 елементів.

Якщо потрібні більш об'ємні таблиці і діаграми, краще підготувати їх для роздаткових матеріалів з допомогою інших програм.

2. Вибрані засоби візуалізації повинні бути адекватними змісту

2.1. Інформація, яка погано сприймається на слух — дати, імена, нові терміни, назви — обов'язково повинна бути представлена на слайдах.

2.2. При графічному поданні інформації повинні використовуватися адекватні засоби візуалізації, тобто підбиратися відповідні змісту типи графіків і діаграм, ілюстрації, таблиці.

2.3. Засоби динамічного представлення інформації (переміщення або різночасне поява фрагментів тексту і графічних об'єктів, інші анімаційні ефекти, ефекти зміни слайдів) повинні служити для дозування інформації, залучення уваги слухачів до тієї її частини, про яку йде мова в певний момент виступу, і показу явищ в динаміці.

3. Кількість і зміст слайдів повинні бути адекватні змісту і тривалості виступу

3.1. Кількість слайдів визначається регламентом виступу. Наприклад, для 10-хвилинного виступу готується не більше 12 слайдів. Надмірна кількість слайдів призводить не тільки до порушення регламенту, але і до стомлення слухачів і розсіювання їх уваги. В зарубіжній практиці надмірна кількість слайдів у презентаціях отримав жартівливе назва «PowerPoint Poisoning» («отруєння» PowerPoint).

3.2. Перший слайд презентації повинен містити назву теми виступу, прізвище, ім'я та по батькові доповідача, час і місце виступу, контактну інформацію. Для корпоративних презентацій в оформленні обов'язково використання відповідної символіки.

3.3. Заключний слайд презентації, що містить текст «Дякую за увагу» або «Кінець», навряд чи прийнятний для презентації супроводжує публічний виступ, оскільки завершення показу слайдів ще не є завершенням виступу. Крім того, такі слайди, так само як і слайд «Питання?», дублюють усне повідомлення. Оптимальним варіантом представляється повторення першого слайда в кінці презентації, оскільки це дає можливість ще раз нагадати слухачам тему виступу та ім'я доповідача і перейти до питань, або завершити виступ. Контактну інформацію, представлену на першому та завершальному слайдах, необхідно також вказати в роздаткових матеріалах.

4. Файл презентації необхідно зберігати у форматі «Демонстрація» (слайд-шоу)

Для показу файл презентації необхідно зберегти у форматі «Демонстрація PowerPoint» (Файл — Зберегти як — Тип файлу — Демонстрація PowerPoint). В цьому випадку презентація автоматично відкривається в режимі повноекранного показу (slideshow) і слухачі позбавлені як від виду робочого вікна програми PowerPoint, так і від втрат часу на початку показу презентації. Збереження презентації у форматі «Демонстрація PowerPoint»

Піктограми файлів у форматах «Демонстрація» (Microsoft PowerPoint Slide Show — *.pps) та «Презентація» (Microsoft PowerPoint Presentation — *.ppt). Окремим аспектом підготовки ілюстративних матеріалів до публічного виступу є створення роздаткових матеріалів. Практика показує, що в підготовці презентацій, що ілюструють публічні виступи, складніше всього: – трансформувати текст виступу для його подання у вигляді ключових слів і фраз, – вибрати адекватні засоби візуалізації інформації, – підібрати оптимальне колірне рішення, – відмовитися від надмірного використання анімаційних ефектів.

Підготовка матеріалів на основі презентацій MS PowerPoint

Роздаткові матеріали є одним із засобів наочності, що використовуються для публічних виступів. Вони можуть надаватися слухачам – до виступу — для попереднього ознайомлення з його змістом; – в процесі виступу — для внесення власних заміток і коментарів;

– після виступу — для повернення до його вмісту в будь-який час.

Крім того, роздаткові матеріали можуть служити і коротким конспектом для самого промовця.

Підготувати роздаткові матеріали можна за допомогою різних програм, у тому числі на основі презентацій MS PowerPoint.

5. Вибір формату

Програма MS PowerPoint дозволяє легко підготувати роздаткові матеріали в різних форматах і варіантах, які вибираються доповідачем у відповідності з його завданнями. У діалоговому вікні «Друк» можна вибрати наступні формати роздруківки презентації:

- слайди;
- власне роздаткові матеріали (в російськомовній версії MS PowerPoint використовуються два варіанти перекладу терміна Handouts — «роздаткові матеріали» і «видачі»);
- структуру презентації;
- нотатки .

При виборі формату «слайди» кожен слайд друкується на окремому аркуші формату А4 в альбомній орієнтації. Використання роздруківки слайдів зручно в тому випадку, якщо в якості матеріалів необхідний фрагмент презентації — наприклад, діаграма або схема, представлена на одному зі слайдів. Для роздруківки окремих слайдів потрібно вказати їх номери в полі «Діапазон друку» у діалоговому вікні «Друк».

При використанні формату «видачі» на одній сторінці можна розмістити 1, 2, 3, 4, 6 або 9 слайдів (на відміну від формату «слайди» в цьому випадку за замовчуванням вибирається книжкова орієнтація паперу).

При цьому на роздаткових матеріалах можуть відображатися:

- тільки слайди;
- слайди і порожні рядки для нотаток.

При виборі формату «структура» на роздаткових матеріалах буде представлений тільки текст слайдів — без графічних елементів.

У форматі «нотатки» кожен слайд буде надрукований на окремому аркуші разом з нотатками виступаючого, які створюються при підготовці презентації в режимі «Звичайний». Цей формат зручно використовувати і в якості конспекту для доповідача.

Під час виступу можна використовувати і електронний варіант слайдів з нотатками на комп'ютері доповідача. Для цього необхідно налаштувати показ на двох екранах (Меню Показ слайдів — На-будівництво презентації — Показ доповідача). В цьому випадку слухачі будуть бачити тільки слайди (Режим показу), а доповідач — слайди, замітки та кнопки керування презентацією (Режим доповідача).

6. Створення колонтитулів

Незалежно від цілей і способу підготовки матеріалів для їх обов'язковим атрибутом є колонтитули. Вони повинні розміщуватися не на слайдах, а на сторінках роздаткових матеріалів. В колонтитули вносяться не тільки номери та кількість сторінок, але і максимально докладна ідентифікуюча інформація, яка дозволяє відновити контекст виступу. До такої інформації відносяться:

- ім'я, прізвище та по батькові промовця;
- назва виступу;
- назва заходу, для якого готувався виступ (семінари, наради, конференції тощо);
- дата виступу;
- контактна інформація (електронна пошта та ін).

Колонтитули на роздаткових матеріалах необхідно підготувати і в тому випадку, якщо презентація буде розміщена у інтернеті, а роздруківку роздавальних матеріалів будуть робити самі користувачі.

Підготувати колонтитули на роздаткових матеріалах в PowerPoint можна в процесі створення презентації (Вид — Колонтитули — Вкладка «Замітки і видачі») і в режимі попереднього перегляду, який можна запустити з пункту меню «Файл» на панелі форматування або за допомогою команди Друк — Перегляд — Друкувати наступне: Видачі; Параметри — Колонтитули — Вкладка «Замітки і видачі» . Побачити

колонтитули в електронній версії матеріалів в MS PowerPoint можна тільки в режимі попереднього перегляду.

Колонтитули для роздавальних матеріалів в MS Word створюються так само, як і для всіх документів MS Word (Вид — Колонтитули).

7. Друк роздаткових матеріалів в MS PowerPoint

Для роздрукування матеріалів в MS PowerPoint необхідно:

1. Відкрити стандартне діалогове вікно «Друк».
2. У випадаючому списку «Друкувати» вибрати необхідний формат: «слайди», «видачі», «нотатки» або «структура».

Якщо планується включення в роздавальні матеріали не всіх слайдів, то в блоці «Діапазон друку» потрібно вказати номери необхідних слайдів.

3. Вибрати кількість слайдів на сторінці, порядок їх слідування (вертикальний або горизонтальний).

4. При роздрукуванні на чорно-білому принтері необхідно в випадаючому списку «Колір або відтінки сірого» вибрати «Відтінки сірого».

Змінити орієнтацію паперу для роздрукування матеріалів можна в діалоговому вікні «Попередній перегляд».

Недоліком роздрукування матеріалів в MS PowerPoint є те, що програма не дозволяє автоматично нумерувати слайди на роздаткових матеріалах. Проставити номери слайдів при необхідності можна перед роздрукуванням у колонтитулах слайдів (Вид — Колонтитули — Слайди або Попередній перегляд — Друкувати — Слайди; Параметри — Колонтитули — Вкладка «Слайди»), але для того щоб номери слайдів були добре видні на роздаткових матеріалах, необхідно у зразку слайдів збільшити розмір шрифту для номера слайда (Меню — Вид — Зразок слайдів — Область номери).

8. Друк роздаткових матеріалів в MS Word

Роздрукувати слайди, структуру презентації або нотатки можна також перенесення презентації у MS Word. Це дає можливість використовувати додаткові формати подання матеріалів: один слайд на сторінці з порожніми рядками для заміток і три слайда на аркуші з текстом нотаток.

Для роздрукування матеріалів в MS Word необхідно:

1. Відкрити презентацію в MS PowerPoint.
2. В меню «Файл» вибрати команду «Відправити» і далі пункт «Microsoft Office Word».
3. У діалоговому вікні «Відправка в Microsoft Office Word» вибрати необхідний формат роздаткових матеріалів.

Якщо при відправленні слайдів у MS Word встановити опцію «Зв'язати», то в документі MS Word будуть автоматично відображатися всі зміни, які будуть вноситися пізніше в презентацію.

4. Після натискання на кнопку «ОК» буде автоматично створений файл MS Word з роздатковими матеріалами.

5. Оформити колонтитули (Меню — Вид — Колонтитули).

6. Зберегти документ під потрібним ім'ям (наприклад, «Роздаткові матеріали»).

При перенесення презентації в MS Word слайди автоматично вставляються в звичайну таблицю MS Word, тому роздаткові матеріали у такому форматі легко редагуються.

Недоліком цього способу підготовки матеріалів є те, що при перенесенні з MS PowerPoint презентацій з великою кількістю графіки файл MS Word виходить дуже об'ємним.

При створенні матеріалів треба також пам'ятати про те, що:

– немає необхідності включати в роздавальні матеріали останній (однаковий з титульним) слайд презентації MS PowerPoint. Наявність такого слайда виправдано в презентації, що супроводжує публічний виступ; у роздавальних ж матеріалах його зміст є надлишковим. Останній слайд можна виключити з матеріалів, не вказуючи його номер при роздрукуванні в MS PowerPoint або видаливши його з таблиці в MS Word;

– ілюстративні матеріали з великим об'ємом інформації — складні таблиці, схеми, списки літератури, приклади фрагментів текстів чи обчислень — доцільно підготувати з допомогою інших програм.

Вибір обсягу матеріалів, часу їх подання (до, після, під час виступу) і програмних засобів для їх підготовки цілком визначається змістом виступу, специфікою ілюстративного матеріалу та завданнями виступу.

1. Типи презентацій

- 1.1. Захист дипломної роботи
- 1.2. Захист дисертації
- 1.3. Конференція
- 1.4. Бізнес-презентація для клієнта (замовника)
- 1.5. Бізнес-презентація для партнера
- 1.6. Виступ на семінарі
- 1.7. Лекція

2. Рекомендації

- 2.1. Поради початківцям
- 2.2. Скільки приблизно часу піде на підготовку презентації?
- 2.3. Ієрархічна модель мислення
- 2.4. Тонкі прийоми

3. Етапи підготовки презентації

4. Справа техніки

- 4.1. MS PowerPoint
- 4.2. MS Word
- 4.3. LaTeX-івський пакет beamer

Наведені нижче рекомендації по підготовці виступів і презентацій адресовано, головним чином, студентам і аспірантам.

Типи презентацій

Перш ніж приступати до підготовки презентації, необхідно визначити цільову аудиторію і тривалість виступу. Від цього буде залежати вся побудова презентації. Презентація однієї і тієї ж роботи на 5 хвилин і на півтори години — це дві різні презентації; перша не робиться простим викиданням слайдів з другої.

Розглянемо найбільш типові приватні випадки.

Захист дипломної роботи

Це найбільш коротка презентація, 5-10 хвилин. Необхідно зосередитися на постановці завдання і перерахування власних результатів. Атестаційної комісії має стати зрозуміло: була якась складна проблема, Ви старанно попрацювали над її вирішенням, і що вдалося поліпшити. Члени комісії можуть погано розбиратися в специфіці і навіть термінології саме Вашої роботи, тому не треба говорити про технічні деталі. У той же час, на слайдах ці деталі можуть бути показані у вигляді досить складних формул, таблиць, графіків. Рекомендується кольором або жирним шрифтом виділяти ті ключові фрагменти, на яких Ви зупиняєтеся при обговоренні. Наприклад, Ви говорите: «у цьому функціоналі введено штрафний доданок, що дозволяє...». Штрафний доданок можна виділити кольором; але пояснювати саму формулу — зайва деталізація і марна трата дорогоцінного часу. Якщо вона когось зацікавить, то він

задасть питання або звернеться за деталями до основного тексту роботи. На захистах обов'язково треба підкреслювати, що саме в роботі запропоновано особисто автором. З іншого боку, тут не місце для зайвого самовираження. Вітається строгий стиль презентації (оптимально — чорний на білому тлі) і чітка мова без жодного зайвого слова.

Захист дисертації

Кандидатська — 15-20 хвилин, докторської — 30-45 хвилин. Здавалося б, з'являється час на те, щоб зупинитися на деяких технічних деталях. Однак, і обсяг роботи істотно більше. Про кожну вирішеної задачу доводиться говорити так само конспективно: постановка, навіщо потрібна, що вийшло. Тільки в самих цікавих випадках, якими автор пишається особливо, можна сказати приблизно наступне: «Ідея методу (докази, алгоритму) полягає в тому, щоб...». Чим вище ранг захисту, тим вище вимоги до актуальності вирішених завдань, новизні і нетривіальності передаваних методів рішення. Тому обов'язково треба підкреслювати такі деталі: скільки років ця проблема залишалася відкритою до Вас, які можливості відкриває її рішення, у скільки разів зросла якість рішення, і т. п. Зрозуміло, самовираження також не бажано.

Конференція

Типове виступ на конференції займає 20 хвилин. Це оптимальний час для того, щоб розповісти про одному завершеному дослідженні, встигнувши розкрити найбільш важливі технічні деталі. Презентація орієнтована на підготовленого слухача, який вже в темі. Тому надмірна популяризація та вступні фрази про користь корисного недоречні. Ступінь деталізації повинна визначатися тим, скільки людина (з Вашої суб'єктивної оцінки) здатні зрозуміти Ваші пояснення. Бажано, щоб кожен слайд і кожна ідея були зрозумілі хоча б половиною аудиторії. Пара жартів протягом доповіді, що підкреслюють Вашу індивідуальність і вносять деяке пожвавлення, цілком доречні. Дизайн слайдів має бути суворим; ніщо не повинно відволікати від розуміння суті самої роботи.

Бізнес-презентація для клієнта (замовника)

Це складний жанр, так як слухач, швидше за все, буде максимально залежною. Тривалість презентації та складу слухачів, як правило, відомі заздалегідь. Тому презентація повинна бути максимально точно розрахована на ступінь підготовленості конкретних людей. Подача матеріалу на слайдах повинна бути професійною з точки зору дизайну. Діаграми, схеми, графіки та інші види ділової графіки вітаються. Слайди повинні демонструвати, що Ваша компанія накопичила величезний професійний досвід, розповісти про який в одній презентації просто немислимо. Зайве наукообразие не привертатиме уваги. Формули припустимі лише у крайніх випадках (наприклад, якщо ви продаєте методику розрахунку прибутковості). Мова повинна бути максимально доступною; враження, що Ви хочете заплутати слухача, рівносильно провалу.

Бізнес-презентація для партнера

Тривалість презентації — договірна. Передбачається, що слухач налаштований дружньо, і, швидше за все, буде вникати у всі деталі і задавати питання, поки не досягне повної ясності. Хорошим тоном в цій ситуації є економія часу Вашого партнера. Тому слайди і структуру виступу необхідно ретельно продумати так, щоб провести слухача крок за кроком, від простого до складного, з усіх деталей презентованого проекту. Тут недоречні міркування про користь корисного, так як в загальних питаннях партнер, швидше за все, знав не гірше за Вас.

Виступ на семінарі

Стиль презентації залежить від того, скільки часу Вам виділено. Це щось середнє між виступом на конференції та презентацією для партнера. Слухач максимально дружній. Ваша мета — досягти повного розуміння у всієї аудиторії. Тому презентація повинна плавно переходити від популярного введення до більш складним технічним деталям. Дизайн слайдів

повинен бути простим і строгим (оптимально — темний на білому); ніщо не повинно відволікати від розуміння суті роботи.

Лекція

У типовому випадку півтори години. Виступаючий має максимальну свободу вибору засобів для розкриття теми; але й утримувати увагу аудиторії протягом тривалого часу набагато складніше. Щоб слухачі не втратили живий інтерес до предмета, на самому початку непогано пояснити, навіщо все це потрібно, коротко розповісти про області застосування, і протягом всієї лекції час від часу повертатися до цих прикладів, демонструючи зв'язок теорії і практики. Дизайн слайдів повинен бути простим і строгим (оптимально — темний на білому); ніщо не повинно відволікати від розуміння змісту лекції.

Рекомендації

Поради початківцям

- Якщо Ви відчуваєте себе хоч трохи невпевнено перед аудиторією, запишіть і вивчіть свою промову напам'ять. Запис виступу на 7 хвилин займає приблизно півтори сторінки тексту (формат A4, шрифт 12pt).

- Має сенс бути акуратним. Неохайно зроблені слайди (різної у шрифтах і відступи, помилки, поліграфічні помилки у формулах) викликають підозру, що і до змістовних питань доповідач підійшов абияк.

- Титульна сторінка необхідна, щоб представити аудиторії Вас і тему Вашої доповіді. На захистах необхідно також вказувати прізвище та ініціали наукового керівника і організацію. На конференціях — назву та дату конференції. Це робиться в тому числі для того, щоб при викладанні в Інтернеті призначення презентації було зрозуміло без додаткових коментарів.

- Оптимальне число рядків на слайді — від 5 до 9. Перевантаженість і дрібний шрифт важкі для сприйняття. Недовантаження залишає враження, що виступ поверхово й погано підготовлено.

- Поширена помилка — читати слайд дослівно. Краще всього, якщо на слайді буде написана детальна інформація (означення, теореми, формули), а словами розповідатиметься їх змістовний сенс. Інформація на слайді може бути більш формальної і строго, викладеної вище, ніж у мові.

- Пункти переліків повинні бути короткими фразами; максимум — два рядки на фразу, оптимально — один рядок. Читання довгої фрази відволікає увагу від мови. Коротка фраза легше запам'ятовується візуально.

- Не промовляйте формули словами — це довго і шалено нудно. Це робиться тільки під час лекцій чи семінарів, коли слухачі одночасно записують конспект. На захисті чи на конференції це недоречно.

- Оптимальна швидкість перемикавання — один слайд за 1-2 хвилини. Для коротких виступів допустимо два слайди у хвилину, але не швидше. Слухачі повинні встигнути сприйняти інформацію і зі слайда, і на слух.

- На слайдах з ключовими визначеннями можна затриматися подовше. Якщо вони не будуть зрозумілі, то не буде зрозуміло нічого.

- Слайди з графіками результатів, навпаки, легко проскакувати в прискореному темпі. Пояснення графіка в типовому випадку: «По горизонтальній осі відкладено ..., по вертикальній осі — ..., видно, що...».

- При поясненні таблиць необхідно говорити, чому відповідають рядки, а чого — стовпці.

- Вводите тільки ті позначення і поняття, без яких розуміння основних ідей доповіді неможливо. Будь-яке позначення повинно бути пояснено до його першого використання (як і в статтях). Якщо пояснення деякого результату вимагає ланцюжки з 20 визначень, то необхідно знайти спосіб пояснити це коротше.

- Громіздкі позначення треба всіляко спрощувати, позбавляючись від зайвих індексів, кришечок, тільдочек, і т. п. Ніхто не змушує Вас дотримуватися тих же позначень, які введені в статті або дисертації. Там напевно це виправдано, а тут треба бути простіше.

- У короткому виступі не можна повторювати одну і ту ж думку, нехай навіть іншими словами — час дорогий. В лекції, навпаки, можна (і потрібно) повертатися до найбільш важливих ідей, розглядаючи їх з нових точок зору. Як говорив Фейнман у своїх лекціях з фізики «ми розуміємо явище, якщо знаходимо йому кілька різних пояснень».

- Над кожною фразою треба критично подумати: чи зрозуміють її слухачі; чи достатньо у них спеціальних знань, щоб її зрозуміти? Незрозумілі фрази слід нещадно вилучати з презентації.

- Будь-яка фраза має говоритися навіщо-то. Не просто тому, що Ви цим займалися в процесі роботи. Кожна фраза повинна логічно підводити до наступних фраз, бути для них посилкою, і в кінцевому підсумку все виступ має бути підпорядковане головній меті — донести до аудиторії дві-три по-справжньому цінних думки. Тільки тоді виступ буде зв'язковим і залишить гарне враження.

Скільки приблизно часу піде на підготовку презентації?

Зрозуміло, це залежить від тривалості виступу (числа слайдів) і ступеня готовності ілюстративного матеріалу.

- Припустимо, що мова йде про захист випускної роботи; матеріал готовий, але презентацію треба цілком складати «з нуля», і досвіду у Вас немає. Тоді в типовому випадку доведеться витратити один або два дні до обговорення з шефом і ще один день на переробку всього після обговорення. Коротше, три дні запланувати потрібно.

- Не треба думати, що гарну презентацію можна зробити за три години напередодні виступу. Це пахне провалом.

- Готову презентацію треба уважно переглянути сім разів, кожен раз будете знаходити за кілька орфографічних помилок або «некрасивостей».

Ієрархічна модель мислення

Людське мислення ієрархічно, або, якщо завгодно, фрактально. Намагаючись зрозуміти тему, людина виділяє кілька ідей як найголовніші (більше 5-7 одночасно в голові не втримати, реально — 3-4). Щоб кожну з них зрозуміти глибше, він розбиває їх на більш прості під-ідеї; ті в свою чергу на ще більш прості ідеї. І так далі. Всю ієрархію ідей в голові не утримати, тому де-то вона неминуче обрізається. При цьому дрібні технічні подробиці так і залишаються незрозумілими, проте без особливого збитку для цілісного сприйняття всієї картини. Вся ця робота проводиться кожним слухачем індивідуально і полуосознанно у процесі сприйняття виступу. По суті справи, виступ — це передача «ієрархії ідей» від людини до людини. Багато принципів ясного викладу спрямовані на те, щоб передавати цю ієрархію в найбільш явному вигляді. Слухач не повинен витрачати своїх ментальних зусиль на відновлення ієрархії ідей з неструктурованого потоку фраз. Це досягається різними технічними прийомами:

- Назва доповіді на першому слайді має відображати саму головну ідею, тобто відповідати вершині ієрархії.

- На початку виступу бажано привести зміст доповіді. Оптимально — три пункти, у кожному — три підпункти. Це структура роботи, два верхніх рівня ієрархії. Якщо виступ короткий, проговорюється тільки верхній рівень.

- Найбільш складні ділянки виступу краще розбивати на кроків, передуючи їх дуже простим слайд зі списком кроків. Наприклад: «Пропонований метод полягає у виконанні трьох кроків:...» або «Є три причини, за якими стандартний підхід не є оптимальним. Розглянемо кожну з них окремо». Якщо цих допоміжних фраз не говорити, зміст доповіді не зміниться; проте сприйматися він буде набагато важче.

- Важливі ідеї верхніх рівнів обов'язково повинні бути відображені на слайдах і сказано словами. Якщо ідея «проскакує» тільки на слайді — це вірна ознака її другорядності.

- Протягом виступу можна кілька разів сказати «основна ідея в тому, щоб...». Але зловживати цією фразою не варто.

- Найбільш важливі слова, ідеї, фрагменти можна виділяти на слайді, щоб вони відразу кидалися в очі. Цим теж не можна зловживати.

Тонкі прийоми

- Мова і слайди не повинні збігатися, тоді презентація стане «об'ємною». Мова повинна бути більш популярна і подібний. Слайди повинні містити більше технічних подробиць: формули, схеми, таблиці, графіки. У короткому виступі в них можна тикати по ходу викладу, але при цьому не треба зупинятися на поясненні всіх дрібниць.

- Перші ж фрази повинні інтригувати. Наприклад, можна сказати про те, наскільки складною або наскільки важливою є дана завдання, або про те, наскільки несподіваним буде рішення — це дозволить утримати увагу слухачів до кінця. Але тоді кінцівка справді повинна виявитися нетривіальною — інакше слухач буде розчарований.

- Останній слайд з висновками: буде краще, якщо висновків буде побільше. На захистах це посилює враження, що пророблена велика робота. Однак проговорюється в кращому випадку тільки пара найбільш важливих висновків; зазвичай просто .

Етапи підготовки презентації:

1. складання плану презентації, виділення основних ідей першого та другого рівня(структура наукової презентації приблизно така ж, як і структура наукової статті);
2. постановка задачі;
3. відомі раніше результати і проблеми;
4. критерії, за яким передбачається оцінювати якість рішення;
5. цілі даної роботи;
6. основні результати автора;
7. умови і результати експериментів;
8. на останньому слайді — перерахування основних результатів роботи.
9. продумування кожного слайда (на перших порах це можна робити вручну на папері), при цьому важливо відповісти на питання:
 10. як ідея цього слайда розкриває основну ідею всієї презентації?
 11. що буде на слайді?
 12. що буде говоритися?
 13. як буде зроблений перехід до наступного слайда?
 14. виготовлення презентації з допомогою однієї з перерахованих нижче систем.

Засоби для створення презентацій

MS PowerPoint - найбільш частий вибір. Якщо на слайді багато формул, рекомендується набирати його повністю в *MS Word* (інакше формули доводиться розміщувати і вирівнювати на слайді вручну). Для цього зручно зробити заготовку — порожній слайд з одним великим Word-об'єктом «Вставка / Об'єкт / Документ Microsoft Word», підібрати один раз його розміри і розмножити на потрібну кількість слайдів. Основний шрифт у тексті і формулах рекомендується змінити на Arial або йому подібний; шрифт Times погано виглядає здалеку. Обов'язково встановіть MathType основний розмір шрифту рівним основного розміру шрифту в тексті. Ніколи не вирівнюйте розмір формули вручну, витягаючи її куточок. В серйозних наукових презентаціях не слід використовувати ефекти анімації. Це дешевий засіб залучення уваги. Колеги на конференціях і члени комісії на захистах можуть образитися, що до них це застосовують...

MS Word на крайній випадок теж піде. Виберіть орієнтацію сторінки landscape, перейдіть в повноекранний режим, встановіть масштаб «По ширині сторінки, перейдіть в

режим «Вид / На весь екран» і під час презентації гортайте сторінки. Цей варіант має суто технічні недоліки: на деяких проекторах можуть з'явитися паразитні ефекти на краях; заважає миготливий курсор; перегортання легко збивається, і тоді слайд стає видно не цілком. Ці недоліки легко усуваються, якщо перевести документ у формат PDF і під час презентації використовувати повноекранний режим Acrobat Reader.

LaTeX-івський пакет beamer – це безсумнівно, це найбільш професійний вибір. LaTeX-івський пакет beamer на порядок перевершує засоби Microsoft Office за якістю результату. Рекомендується освоїти його всім, хто збирається займатися наукою і далі — на конференціях і захистах beamer'ські презентації справляють дуже гарне враження. Презентація стандартним чином переводиться у формат PDF і показується в повноекранному режимі за допомогою Acrobat Reader.

Література: [дод. 4].

Список рекомендованої літератури

Базова

1. Информатика. Базовый курс. 2-е издание / Под ред. С. В. Симоновича. — СПб. : Питер, 2005. — 640 с.
2. Сулима И.М., Гавриленко С.И. и др. Основные численные методы и их реализация на ЭВМ. — К.: Вища школа, 1987 г. — 312 с.
3. Криницкий Н.А. Алгоритмы вокруг нас. — М.: Наука, 1987. — 224 с.
4. Сафронов И.К. Задачник-практикум по информатике. — СПб.: БХВ, 2002. — 492 с.
5. Єдине інформаційне середовище Національного технічного університету України «Київський політехнічний інститут» СИСТЕМА ЕЛЕКТРОННИЙ КАМПУС НТУУ «КПІ» Інструкція користувача Профіль «Викладач-науковець» (ІК 80.3 – 02070921. 004.002: 2011) версія 1.02 – К.: НТУУ «КПІ», 2012. — 33 с.

Допоміжна

1. Задачник-практикум / Под ред. Семакина И.В., Хеннера А.В. — М.: ЭКОМ, 1999. — 304 с.
2. Дьяконов В.П., Абраменкова И.В. MathCAD 7.0 в математике, физике и в Internet. — М.: “Нолидж”, 1999. — 352с.
3. Гарнаев, А. Ю. Excel, VBA, Internet в экономике и финансах.— СПб.: БХВ-Петербург, 2001. — 816с.:ил.
4. Цаповська Жаннета Ярославівна. Робота з Microsoft PowerPoint 2000/ 2003/ 2007: навч. посіб. / Львівський національний ун-т ім. Івана Франка. — Л. : ЛНУ ім. І.Франка, 2009. — 316с.